

---

# Nipyapi Documentation

*Release 0.19.1*

**Daniel Chaffelson**

**Aug 08, 2022**



---

## Contents

---

<b>1</b>	<b>NiPyApi</b>	<b>3</b>
1.1	Features . . . . .	3
1.2	Quick Start . . . . .	4
1.3	Background and Documentation . . . . .	4
1.4	NiFi Version Support . . . . .	4
1.5	Python Requirements . . . . .	4
<b>2</b>	<b>Installation</b>	<b>7</b>
2.1	Stable release . . . . .	7
2.2	From sources . . . . .	7
<b>3</b>	<b>NiPyApi Package Reference</b>	<b>9</b>
3.1	Demos . . . . .	9
3.1.1	FDLC . . . . .	9
3.1.2	Console . . . . .	9
3.1.3	Secured Connection . . . . .	9
3.1.4	bbende How Do I Deploy My Flow . . . . .	10
3.2	Client SDK modules . . . . .	10
3.2.1	Canvas . . . . .	10
3.2.2	Config . . . . .	19
3.2.3	Parameters . . . . .	19
3.2.4	Security . . . . .	21
3.2.5	System . . . . .	25
3.2.6	Templates . . . . .	26
3.2.7	Utils . . . . .	28
3.2.8	Versioning . . . . .	31
3.3	Swagger Client SDKs . . . . .	36
3.3.1	NiFi Swagger Client . . . . .	36
3.3.1.1	Subpackages . . . . .	36
3.3.1.2	Submodules . . . . .	533
3.3.1.3	nipyapi.nifi.api_client module . . . . .	533
3.3.1.4	nipyapi.nifi.configuration module . . . . .	535
3.3.1.5	nipyapi.nifi.rest module . . . . .	536
3.3.2	NiFi-Registry Swagger Client . . . . .	537
3.3.2.1	Subpackages . . . . .	537
3.3.2.2	Submodules . . . . .	619
3.3.2.3	nipyapi.registry.api_client module . . . . .	619

3.3.2.4	nipyapi.registry.configuration module . . . . .	621
3.3.2.5	nipyapi.registry.rest module . . . . .	621
<b>4</b>	<b>ToDo</b>	<b>623</b>
<b>5</b>	<b>Contributing</b>	<b>625</b>
5.1	Types of Contributions . . . . .	625
5.1.1	Report Bugs . . . . .	625
5.1.2	Fix Bugs . . . . .	625
5.1.3	Implement Features . . . . .	625
5.1.4	Write Documentation . . . . .	626
5.1.5	Submit Feedback . . . . .	626
5.2	Get Started! . . . . .	626
5.3	Pull Request Guidelines . . . . .	627
<b>6</b>	<b>Development Notes</b>	<b>629</b>
6.1	Decision Points . . . . .	629
6.2	Testing Notes . . . . .	629
6.3	Docker Test Environment . . . . .	629
6.4	Remote Testing on Centos7 . . . . .	629
6.5	Testing on OSX . . . . .	630
6.6	Generate Swagger Client . . . . .	630
6.6.1	NiFi Swagger Client . . . . .	630
6.6.2	NiFi Registry Swagger Client . . . . .	631
6.7	Release Process . . . . .	631
<b>7</b>	<b>Credits</b>	<b>633</b>
7.1	Development Lead . . . . .	633
7.2	Contributors . . . . .	633
7.3	Shout Outs . . . . .	633
<b>8</b>	<b>History</b>	<b>635</b>
8.1	0.19.1 (2022-08-08) . . . . .	635
8.2	0.19.0 (2022-05-13) . . . . .	635
8.3	0.18.0 (2021-11-29) . . . . .	636
8.4	0.17.1 (2021-10-21) . . . . .	636
8.5	0.17.0 (2021-10-13) . . . . .	636
8.6	0.16.3 (2021-10-11) . . . . .	636
8.7	0.16.2 (2021-02-10) . . . . .	636
8.8	0.16.3 (2021-10-11) . . . . .	637
8.9	0.16.2 (2021-02-10) . . . . .	638
8.10	0.15.0 (2020-11-06) . . . . .	638
8.11	0.14.0 (2019-11-06) . . . . .	639
8.12	0.13.3 (2019-10-09) . . . . .	639
8.13	0.13.0 (2019-04-22) . . . . .	639
8.14	0.12.0 (2018-12-20) . . . . .	639
8.15	0.11.0 (2018-10-12) . . . . .	640
8.16	0.10.3 (2018-08-28) . . . . .	640
8.17	0.10.2 (2018-08-27) . . . . .	641
8.18	0.10.1 (2018-08-21) . . . . .	641
8.19	0.10.0 (2018-08-03) . . . . .	641
8.20	0.9.1 (2018-05-18) . . . . .	641
8.21	0.9.0 (2018-05-16) . . . . .	641
8.22	0.8.0 (2018-03-06) . . . . .	642
8.23	0.7.0 (2018-01-30) . . . . .	643

8.24	0.6.1 (2018-01-04)	644
8.25	0.6.0 (2017-12-31)	644
8.26	0.5.1 (2017-12-07)	644
8.27	0.5.0 (2017-12-06)	644
8.28	0.4.0 (2017-10-29)	644
8.29	0.3.2 (2017-09-04)	645
8.30	0.3.1 (2017-09-04)	645
8.31	0.3.0 (2017-09-04)	645
8.32	0.2.1 (2017-08-26)	645
8.33	0.2.0 (2017-08-25)	645
8.34	0.1.2 (2017-08-24)	645
8.35	0.1.1 (2017-08-24)	646
8.36	0.1.0 (2017-08-24)	646
<b>9</b>	<b>Indices and tables</b>	<b>647</b>
	<b>Python Module Index</b>	<b>649</b>
	<b>Index</b>	<b>655</b>



Contents:





Nifi-Python-API: A rich Apache NiFi Python Client SDK

## 1.1 Features

**Three layers of Python support for working with Apache NiFi:**

- High-level Demos and example scripts
- Mid-level Client SDK for typical complex tasks
- Low-level Client SDKs for the full API implementation of NiFi and selected sub-projects

**Functionality Highlights:**

- Detailed documentation of the full SDK at all levels
- CRUD wrappers for common task areas like Processor Groups, Processors, Templates, Registry Clients, Registry Buckets, Registry Flows, etc.
- Convenience functions for inventory tasks, such as recursively retrieving the entire canvas, or a flat list of all Process Groups
- Support for scheduling and purging flows, controller services, and connections
- Support for fetching and updating Variable Registries
- Support for import/export of Versioned Flows from NiFi-Registry
- Docker Compose configurations for testing and deployment
- A scripted deployment of an interactive environment, and a secured configuration, for testing and demonstration purposes

Please see the [issue](#) register for more information on current development.

## 1.2 Quick Start

There are several scripts to produce demo environments in *nipyapi.demo.\**

The mid-level functionality is in *nipyapi.canvas* / *nipyapi.security* / *nipyapi.templates* / *nipyapi.versioning*

You can access the entire API using the low-level SDKs in *nipyapi.nifi* / *nipyapi.registry*

The easiest way to install NiPyApi is with pip:

```
# in bash
pip install nipyapi
```

You can set the config for your endpoints in the central config file:

```
# in python
import nipyapi
nipyapi.config.nifi_config.host = 'http://localhost:8080/nifi-api'
nipyapi.config.registry_config.host = 'http://localhost:18080/nifi-registry-api'
```

Then import a module and execute tasks:

```
nipyapi.canvas.get_root_pg_id()
>'4d5dcf9a-015e-1000-097e-e505ed0f7fd2'
```

You can use the Docker demos to create a secured interactive console showing many features:

```
from nipyapi.demo.secured_console import *
from nipyapi.demo.console import *
```

You can also explore the scripts to get ideas for how NiPyAPI can be used to automate your environment.

Please check out the [Contribution Guide](#) if you are interested in contributing to the feature set.

## 1.3 Background and Documentation

For more information on Apache NiFi, please visit <https://nifi.apache.org>

For Documentation on this package please visit <https://nipyapi.readthedocs.io>.

## 1.4 NiFi Version Support

Currently we are testing against NiFi versions 1.1.2 - 1.15.0, and NiFi-Registry versions 0.1.0 - 1.15.0.

If you find a version compatibility problem please raise an [issue](#)

## 1.5 Python Requirements

Python 2.7 or 3.6-9 supported, though other versions may work. *We will shortly stop supporting Python2 There are known issues on Python 3.10*

Tested on AL2 and OSX 10.14.x - Windows automated testing not attempted

Outside of the standard Python modules, we make use of lxml, DeepDiff, ruamel.yaml and xmldict in processing, and Docker for demo/tests.



### 2.1 Stable release

To install Nipyapi, run this command in your terminal:

```
$ pip install nipyapi
```

This is the preferred method to install Nipyapi, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

### 2.2 From sources

The sources for Nipyapi can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/Chaffelson/nipyapi
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/Chaffelson/nipyapi/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```



### 3.1 Demos

These modules leverage functionality within the rest of the Package to demonstrate various capabilities

#### 3.1.1 FDLC

Importing this module provides further instructions for it's use. It will guide you through the steps involved in flow promotion.

Note that it makes extensive use of Docker Containers.

Usage:

```
from nipyapi.demo.fdlc import *
```

#### 3.1.2 Console

Importing this module will run a script which populates the NiFi canvas with a Process Group containing a processor, and creates a sequence of Versioned Flow Objects from it, along with a Template and various export versions.

This is intended to give the user a base set of objects to explore the API.

Usage:

```
from nipyapi.demo.console import *
```

#### 3.1.3 Secured Connection

Importing this module will pull recent Docker containers from Dockerhub, deploy them in a secured configuration, and prepare the environment for access via TLS in NiFi-Registry's case, and public LDAP username/password for

NiFi.

This is intended to give the user an example of a secured environment. May be combined with the Console to produce a secured environment with demo objects.

Note that this demo makes extensive use of Docker Containers.

Usage:

```
from nipyapi.demo.secured_connection import *
```

### 3.1.4 bbende How Do I Deploy My Flow

An incomplete version of BBende's excellent demo. It currently deploys some Docker NiFi and Registry instances.

## 3.2 Client SDK modules

These wrapper modules contain collections of convenience functions for daily operations of your NiFi and NiFi-Registry environment. They wrap and surface underlying data structures and calls to the full SDK swagger clients which are also included in the package.

### 3.2.1 Canvas

For interactions with the NiFi Canvas.

`nipyapi.canvas.get_root_pg_id()`

Convenience function to return the UUID of the Root Process Group

Returns (str): The UUID of the root PG

`nipyapi.canvas.recurse_flow(pg_id='root')`

Returns information about a Process Group and all its Child Flows. Recurses the child flows by appending each process group with a 'nipyapi\_extended' parameter which contains the child process groups, etc. Note: This previously used actual recursion which broke on large NiFi

environments, we now use a task/list update approach

**Parameters** `pg_id (str)` – The Process Group UUID

**Returns** enriched NiFi Flow object

**Return type** (*ProcessGroupFlowEntity*)

`nipyapi.canvas.get_flow(pg_id='root')`

Returns information about a Process Group and flow.

This surfaces the native implementation, for the recursed implementation see 'recurse\_flow'

**Parameters** `pg_id (str)` – id of the Process Group to retrieve, defaults to the root process group if not set

**Returns** The Process Group object

**Return type** (*ProcessGroupFlowEntity*)



`nipyapi.canvas.get_process_group_status (pg_id='root', detail='names')`

Returns an entity containing the status of the Process Group. Optionally may be configured to return a simple dict of name:id pairings

Note that there is also a 'process group status' command available, but it returns a subset of this data anyway, and this call is more useful

#### Parameters

- **pg\_id** (*str*) – The UUID of the Process Group
- **detail** (*str*) – 'names' or 'all'; whether to return a simple dict of name:id pairings, or the full details. Defaults to 'names'

**Returns** The Process Group Entity including the status

**Return type** (*ProcessGroupEntity*)

`nipyapi.canvas.get_process_group (identifier, identifier_type='name', greedy=True)`

Filters the list of all process groups against a given identifier string occurring in a given identifier\_type field.

#### Parameters

- **identifier** (*str*) – the string to filter the list for
- **identifier\_type** (*str*) – the field to filter on, set in config.py
- **greedy** (*bool*) – True for partial match, False for exact match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.canvas.list_all_process_groups (pg_id='root')`

Returns a flattened list of all Process Groups on the canvas. Potentially slow if you have a large canvas.

Note that the `ProcessGroupsApi().get_process_groups(pg_id)` command only provides the first layer of pgs, whereas this trawls the entire canvas

**Parameters** **pg\_id** (*str*) – The UUID of the Process Group to start from, defaults to the Canvas root

**Returns** list[ProcessGroupEntity]

`nipyapi.canvas.delete_process_group (process_group, force=False, refresh=True)`

Deletes a given Process Group, with optional prejudice.

#### Parameters

- **process\_group** (*ProcessGroupEntity*) – The target Process Group
- **force** (*bool*) – Stop, purge and clean the target Process Group before deletion. Experimental.
- **refresh** (*bool*) – Whether to refresh the state first

**Returns** The updated object state

**Return type** (*ProcessGroupEntity*)

`nipyapi.canvas.schedule_process_group (process_group_id, scheduled)`

Start or Stop a Process Group and all components.

Note that this doesn't guarantee that all components have started, as some may be in Invalid states.

#### Parameters

- **process\_group\_id** (*str*) – The UUID of the target Process Group
- **scheduled** (*bool*) – True to start, False to stop

**Returns** True of successfully scheduled, False if not

**Return type** (bool)

`nipyapi.canvas.create_process_group(parent_pg, new_pg_name, location, comment=)`

Creates a new Process Group with the given name under the provided parent Process Group at the given Location

**Parameters**

- **parent\_pg** (`ProcessGroupEntity`) – The parent Process Group to create the new process group in
- **new\_pg\_name** (`str`) – The name of the new Process Group
- **location** (`tuple[x, y]`) – the x,y coordinates to place the new Process Group under the parent
- **comment** (`str`) – Entry for the Comments field

**Returns** The new Process Group

**Return type** (`ProcessGroupEntity`)

`nipyapi.canvas.list_all_processors(pg_id='root')`

Returns a flat list of all Processors under the provided Process Group

**Parameters** **pg\_id** (`str`) – The UUID of the Process Group to start from, defaults to the Canvas root

**Returns** list[`ProcessorEntity`]

`nipyapi.canvas.list_all_processor_types()`

Produces the list of all available processor types in the NiFi instance

**Returns** A native datatype containing the processors list

**Return type** list(`ProcessorTypesEntity`)

`nipyapi.canvas.get_processor_type(identifier, identifier_type='name', greedy=True)`

Gets the abstract object describing a Processor, or list thereof

**Parameters**

- **identifier** (`str`) – the string to filter the list for
- **identifier\_type** (`str`) – the field to filter on, set in config.py
- **greedy** (`bool`) – False for exact match, True for greedy match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.canvas.create_processor(parent_pg, processor, location, name=None, config=None)`

Instantiates a given processor on the canvas

**Parameters**

- **parent\_pg** (`ProcessGroupEntity`) – The parent Process Group
- **processor** (`DocumentedTypeDTO`) – The abstract processor type object to be instantiated
- **location** (`tuple[x, y]`) – The location coordinates
- **name** (`Optional [str]`) – The name for the new Processor
- **config** (`Optional [ProcessorConfigDTO]`) – A configuration object for the new processor

**Returns** The new Processor

**Return type** (*ProcessorEntity*)

`nipyapi.canvas.delete_processor(processor, refresh=True, force=False)`

Deletes a Processor from the canvas, with optional prejudice.

**Parameters**

- **processor** (*ProcessorEntity*) – The processor to delete
- **refresh** (*bool*) – Whether to refresh the Processor state before action
- **force** (*bool*) – Whether to stop, purge and remove connections to the Processor before deletion. Behavior may change in future releases.

**Returns** The updated ProcessorEntity

**Return type** (*ProcessorEntity*)

`nipyapi.canvas.get_processor(identifier, identifier_type='name', greedy=True)`

Filters the list of all Processors against the given identifier string in the given identifier\_type field

**Parameters**

- **identifier** (*str*) – The String to filter against
- **identifier\_type** (*str*) – The field to apply the filter to. Set in config.py
- **greedy** (*bool*) – Whether to exact match (False) or partial match (True)

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.canvas.schedule_processor(processor, scheduled, refresh=True)`

Set a Processor to Start or Stop.

Note that this doesn't guarantee that it will change state, merely that it will be instructed to try. Some effort is made to wait and see if the processor starts

**Parameters**

- **processor** (*ProcessorEntity*) – The Processor to target
- **scheduled** (*bool*) – True to start, False to stop
- **refresh** (*bool*) – Whether to refresh the object before action

**Returns** True for success, False for failure

**Return type** (*bool*)

`nipyapi.canvas.get_funnel(funnel_id)`

Gets a given Funnel by ID

`nipyapi.canvas.update_processor(processor, update, refresh=True)`

Updates configuration parameters for a given Processor.

An example update would be: `nifi.ProcessorConfigDTO(scheduling_period='3s')`

**Parameters**

- **processor** (*ProcessorEntity*) – The Processor to target for update
- **update** (*ProcessorConfigDTO*) – The new configuration parameters
- **refresh** (*bool*) – Whether to refresh the Processor object state before applying the update

**Returns** The updated ProcessorEntity

**Return type** (*ProcessorEntity*)

`nipyapi.canvas.get_variable_registry(process_group, ancestors=True)`

Gets the contents of the variable registry attached to a Process Group

**Parameters**

- **process\_group** (*ProcessGroupEntity*) – The Process Group to retrieve the Variable Registry from
- **ancestors** (*bool*) – Whether to include the Variable Registries from child Process Groups

**Returns** The Variable Registry

**Return type** (*VariableRegistryEntity*)

`nipyapi.canvas.update_variable_registry(process_group, update, refresh=True)`

Updates one or more key:value pairs in the variable registry

**Parameters**

- **process\_group** (*ProcessGroupEntity*) – The Process Group which has the
- **Registry to be updated** (*Variable*) –
- **update** (*list[tuple]*) – The variables to write to the registry
- **refresh** (*bool*) – Whether to refresh the object revision before updating

**Returns** The created or updated Variable Registry Entries

**Return type** (*VariableRegistryEntity*)

`nipyapi.canvas.purge_connection(con_id)`

EXPERIMENTAL Drops all FlowFiles in a given connection. Waits until the action is complete before returning.

Note that if upstream component isn't stopped, more data may flow into the connection after this action.

**Parameters** **con\_id** (*str*) – The UUID of the Connection to be purged

**Returns** The status reporting object for the drop request.

**Return type** (*DropRequestEntity*)

`nipyapi.canvas.purge_process_group(process_group, stop=False)`

EXPERIMENTAL Purges the connections in a given Process Group of FlowFiles, and optionally stops it first

**Parameters**

- **process\_group** (*ProcessGroupEntity*) – Target Process Group
- **stop** (*Optional [bool]*) – Whether to stop the Process Group before action

**Returns** `True|False}}`: Result set. A list of Dicts of

**Return type** (*list[dict{ID*

Connection IDs mapped to True or False for success of each connection

`nipyapi.canvas.schedule_components(pg_id, scheduled, components=None)`

Changes the scheduled target state of a list of components within a given Process Group.

Note that this does not guarantee that components will be Started or Stopped afterwards, merely that they will have their scheduling updated.

**Parameters**

- **pg\_id** (*str*) – The UUID of the parent Process Group
- **scheduled** (*bool*) – True to start, False to stop
- **components** (*list[ComponentType]*) – The list of Component Entities to schedule, e.g. ProcessorEntity's

**Returns** True for success, False for not

**Return type** (*bool*)

`nipyapi.canvas.get_bulletins()`

Retrieves current bulletins (alerts) from the Flow Canvas

**Returns** The native datatype containing a list

**Return type** (*ControllerBulletinsEntity*)

of bulletins

`nipyapi.canvas.get_bulletin_board()`

Retrieves the bulletin board object

**Returns** The native datatype BulletinBoard object

**Return type** (*BulletinBoardEntity*)

`nipyapi.canvas.list_invalid_processors(pg_id='root', summary=False)`

Returns a flattened list of all Processors with Invalid Statuses

**Parameters**

- **pg\_id** (*str*) – The UUID of the Process Group to start from, defaults to the Canvas root
- **summary** (*bool*) – True to return just the list of relevant properties per Processor, False for the full listing

**Returns** *list[ProcessorEntity]*

`nipyapi.canvas.list_sensitive_processors(pg_id='root', summary=False)`

Returns a flattened list of all Processors on the canvas which have sensitive properties that would need to be managed during deployment

**Parameters**

- **pg\_id** (*str*) – The UUID of the Process Group to start from, defaults to the Canvas root
- **summary** (*bool*) – True to return just the list of relevant properties per Processor, False for the full listing

**Returns** *list[ProcessorEntity]* or *list(dict)*

`nipyapi.canvas.list_all_connections(pg_id='root', descendants=True)`

Lists all connections for a given Process Group ID

**Parameters**

- **pg\_id** (*str*) – ID of the Process Group to retrieve Connections from
- **descendants** (*bool*) – True to recurse child PGs, False to not

**Returns** List of ConnectionEntity objects

**Return type** (*list*)

`nipyapi.canvas.create_connection(source, target, relationships=None, name=None)`

Creates a connection between two objects for the given relationships

**Parameters**

- **source** – Object to initiate the connection, e.g. ProcessorEntity
- **target** – Object to terminate the connection, e.g. FunnelEntity
- **relationships** (*list*) – list of strings of relationships to connect, may be collected from the object ‘relationships’ property (optional)
- **name** (*str*) – Defaults to None, String of Name for Connection (optional)

**Returns** for the created connection

**Return type** (*ConnectionEntity*)

`nipyapi.canvas.delete_connection(connection, purge=False)`

Deletes a connection, optionally purges it first

**Parameters**

- **connection** (*ConnectionEntity*) – Connection to delete
- **purge** (*bool*) – True to Purge, Defaults to False

**Returns** the modified Connection

**Return type** (*ConnectionEntity*)

`nipyapi.canvas.get_component_connections(component)`

Returns list of Connections related to a given Component, e.g. Processor

**Parameters** **component** – Component Object to filter by, e.g. a ProcessorEntity

**Returns** List of ConnectionEntity Objects

**Return type** (*list*)

`nipyapi.canvas.create_controller(parent_pg, controller, name=None)`

**Creates a new Controller Service in a given Process Group of the given** Controller type, with the given Name

**Parameters**

- **parent\_pg** (*ProcessGroupEntity*) – Target Parent PG
- **controller** (*DocumentedTypeDTO*) – Type of Controller to create, found via the `list_all_controller_types` method
- **name** (*str[Optional]*) – Name for the new Controller as a String

**Returns** (*ControllerServiceEntity*)

`nipyapi.canvas.list_all_controllers(pg_id='root', descendants=True)`

**Lists all controllers under a given Process Group, defaults to Root** Optionally recurses all child Process Groups as well

**Parameters**

- **pg\_id** (*str*) – String of the ID of the Process Group to list from
- **descendants** (*bool*) – True to recurse child PGs, False to not

**Returns** None, ControllerServiceEntity, or list(ControllerServiceEntity)

`nipyapi.canvas.delete_controller(controller, force=False)`

Delete a Controller service, with optional prejudice

#### Parameters

- **controller** (`ControllerServiceEntity`) – Target Controller to delete
- **force** (`bool`) – True to attempt Disable the Controller before deletion

**Returns** (`ControllerServiceEntity`)

`nipyapi.canvas.update_controller(controller, update)`

Updates the Configuration of a Controller Service

#### Parameters

- **controller** (`ControllerServiceEntity`) – Target Controller to update
- **update** (`ControllerServiceDTO`) – Controller Service configuration object containing the new config params and properties

**Returns** (`ControllerServiceEntity`)

`nipyapi.canvas.schedule_controller(controller, scheduled, refresh=False)`

Start/Enable or Stop/Disable a Controller Service

#### Parameters

- **controller** (`ControllerServiceEntity`) – Target Controller to schedule
- **scheduled** (`bool`) – True to start, False to stop
- **refresh** (`bool`) – Whether to refresh the component revision before execution

**Returns** (`ControllerServiceEntity`)

`nipyapi.canvas.get_controller(identifier, identifier_type='name', bool_response=False)`

Retrieve a given Controller

#### Parameters

- **identifier** (`str`) – ID or Name of a Controller to find
- **identifier\_type** (`str`) – 'id' or 'name', defaults to name
- **bool\_response** (`bool`) – If True, will return False if the Controller is not found - useful when testing for deletion completion

Returns:

`nipyapi.canvas.list_all_controller_types()`

Lists all Controller Service types available on the environment

**Returns** `list(DocumentedTypeDTO)`

`nipyapi.canvas.list_all_by_kind(kind, pg_id='root', descendants=True)`

Retrieves a list of all instances of a supported object type

#### Parameters

- **kind** (`str`) – one of `input_ports`, `output_ports`, `funnels`, `controllers`, `connections`, `remote_process_groups`
- **pg\_id** (`str`) – optional, ID of the Process Group to use as search base
- **descendants** (`bool`) – optional, whether to collect child group info

**Returns** `list` of the Entity type of the kind, or single instance, or None

`nipyapi.canvas.list_all_input_ports (pg_id='root', descendants=True)`

Convenience wrapper for `list_all_by_kind` for input ports

`nipyapi.canvas.list_all_output_ports (pg_id='root', descendants=True)`

Convenience wrapper for `list_all_by_kind` for output ports

`nipyapi.canvas.list_all_funnels (pg_id='root', descendants=True)`

Convenience wrapper for `list_all_by_kind` for funnels

`nipyapi.canvas.list_all_remote_process_groups (pg_id='root', descendants=True)`

Convenience wrapper for `list_all_by_kind` for remote process groups

`nipyapi.canvas.delete_funnel (funnel, refresh=True)`

Deletes a Funnel Object

#### Parameters

- **funnel** (`FunnelEntity`) – The Funnel to delete
- **refresh** (`bool`) – Whether to refresh the object state before execution

**Returns** (`FunnelEntity`) Deleted `FunnelEntity` reference

`nipyapi.canvas.get_remote_process_group (rpg_id, summary=False)`

Fetch a remote process group object, with optional summary of just ports

`nipyapi.canvas.update_process_group (pg, update, refresh=True)`

Updates a given Process Group.

#### Parameters

- **pg** (`ProcessGroupEntity`) – The Process Group to target for update
- **update** (`dict`) – key:value pairs to update
- **refresh** (`bool`) – Whether to refresh the Process Group before applying the update

**Returns** The updated `ProcessorEntity`

**Return type** (`ProcessGroupEntity`)

`nipyapi.canvas.create_funnel (pg_id, position=None)`

Creates a Funnel Object

#### Parameters

- **pg\_id** (`str`) – ID of the parent Process Group
- **position** (`tuple[int, int]`) – Position on canvas

**Returns** (`FunnelEntity`) Created Funnel

`nipyapi.canvas.create_remote_process_group (target_uris, transport='RAW', pg_id='root', position=None)`

Creates a new Remote Process Group with given parameters

#### Parameters

- **target\_uris** (`str`) – Comma separated list of target URIs
- **transport** (`str`) – optional, RAW or HTTP
- **pg\_id** (`str`) – optional, UUID of parent Process Group for remote process group
- **position** (`tuple`) – optional, tuple of location ints

**Returns** (`RemoteProcessGroupEntity`)



```
nipyapi.canvas.delete_remote_process_group(rpg, refresh=True)
```

Deletes a given remote process group

#### Parameters

- **rpg** (`RemoteProcessGroupEntity`) – Remote Process Group to remove
- **refresh** (`bool`) – Whether to refresh the object before action

**Returns** (`RemoteProcessGroupEntity`)

```
nipyapi.canvas.set_remote_process_group_transmission(rpg, enable=True, re-
                                                    fresh=True)
```

Enable or Disable Transmission for an RPG

#### Parameters

- **rpg** (`RemoteProcessGroupEntity`) – The ID of the remote process group to modify
- **enable** (`bool`) – True to enable, False to disable
- **refresh** (`bool`) – Whether to refresh the object before action

Returns:

```
nipyapi.canvas.get_pg_parents_ids(pg_id)
```

Retrieve the ids of the parent Process Groups.

**Parameters** **pg\_id** (`str`) – Process group id

**Returns** (`list`) List of ids of the input PG parents

```
nipyapi.canvas.delete_port(port)
```

Deletes a given port from the canvas if possible

```
nipyapi.canvas.create_port(pg_id, port_type, name, state, position=None)
```

Creates a new input or output port of given characteristics

#### Parameters

- **pg\_id** (`str`) – ID of the parent Process Group
- **port\_type** (`str`) – Either of INPUT\_PORT or OUTPUT\_PORT
- **name** (`str`) – optional, Name to assign to the port
- **state** (`str`) – One of RUNNING, STOPPED, DISABLED
- **position** (`tuple`) – optional, tuple of ints like (400, 400)

**Returns** (`PortEntity`) of the created port

## 3.2.2 Config

A set of defaults and parameters used elsewhere in the project. Also provides a handy link to the low-level client SDK configuration singleton objects.

## 3.2.3 Parameters

For Managing NiFi Parameter Contexts

```
nipyapi.parameters.list_all_parameter_contexts()
```

Lists all Parameter Contexts available on the Canvas

**Returns** list(ParameterContextEntity)

`nipyapi.parameters.create_parameter_context` (*name*, *description=None*, *parameters=None*, *inherited\_contexts=None*)

Create a new Parameter Context with optional description and initial Parameters

#### Parameters

- **name** (*str*) – The Name for the new Context
- **description** (*str*) – An optional description
- **parameters** (*list[ParameterEntity]*) – A list of prepared Parameters
- **inherited\_contexts** (*list[ParameterContextEntity]*) – A list of inherited Parameter Contexts

**Returns** (ParameterContextEntity) The New Parameter Context

`nipyapi.parameters.delete_parameter_context` (*context*, *refresh=True*)

Removes a Parameter Context

#### Parameters

- **context** (*ParameterContextEntity*) – Parameter Context to be deleted
- **refresh** (*bool*) – Whether to refresh the Context before Deletion

**Returns** (ParameterContextEntity) The removed Parameter Context

`nipyapi.parameters.get_parameter_context` (*identifier*, *identifier\_type='name'*, *greedy=True*)

Gets one or more Parameter Contexts matching a given identifier

#### Parameters

- **identifier** (*str*) – The Name or ID matching Parameter Context(s)
- **identifier\_type** (*str*) – 'name' or 'id'
- **greedy** (*bool*) – False for exact match, True for string match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.parameters.update_parameter_context` (*context*)

Update an already existing Parameter Context

#### Parameters

- **context** (*ParameterContextEntity*) – Parameter Context updated to be applied
- **refresh** (*bool*) – Whether to refresh the object before Updating

**Returns** (ParameterContextEntity) The updated Parameter Context

`nipyapi.parameters.prepare_parameter` (*name*, *value*, *description=None*, *sensitive=False*)

Parses basic inputs into a Parameter object ready for submission

#### Parameters

- **name** (*str*) – The Name for the Parameter
- **value** (*str*, *int*, *float*) – The Value for the Parameter
- **description** (*str*) – Optional Description for the Parameter
- **sensitive** (*bool*) – Whether to mark the Parameter Value as sensitive

**Returns** (ParameterEntity) The ParameterEntity ready for use

`nipyapi.parameters.delete_parameter_from_context(context, parameter_name)`

Delete a specific Parameter from a Parameter Context :param context: The Parameter Context to Update :type context: ParameterContextEntity :param parameter\_name: The Parameter to delete :type parameter\_name: str

**Returns** (ParameterContextEntity) The updated Parameter Context

`nipyapi.parameters.upsert_parameter_to_context(context, parameter)`

Insert or Update Parameter within a Parameter Context

#### Parameters

- **context** (ParameterContextEntity) – The Parameter Context to Modify
- **parameter** (ParameterEntity) – The ParameterEntity to insert or update

**Returns** (ParameterContextEntity) The updated Parameter Context

`nipyapi.parameters.assign_context_to_process_group(pg, context_id, cascade=False)`

Assigns a given Parameter Context to a specific Process Group Optionally cascades down to direct children Process Groups

#### Parameters

- **pg** (ProcessGroupEntity) – The Process Group to target
- **context\_id** (str) – The ID of the Parameter Context
- **cascade** (bool) – Cascade Parameter Context down to child Process Groups?

**Returns** (ProcessGroupEntity) The updated Process Group

`nipyapi.parameters.remove_context_from_process_group(pg)`

Clears any Parameter Context from the given Process Group

**Parameters** **pg** (ProcessGroupEntity) – The Process Group to target

**Returns** (ProcessGroupEntity) The updated Process Group

## 3.2.4 Security

Secure connectivity management for NiPyApi

`nipyapi.security.create_service_user(identity, service='nifi', strict=True)`

Attempts to create a user with the provided identity in the given service

#### Parameters

- **identity** (str) – Identity string for the user
- **service** (str) – 'nifi' or 'registry'
- **strict** (bool) – If Strict, will error if user already exists

**Returns** The new (User) or (UserEntity) object

`nipyapi.security.create_service_user_group(identity, service='nifi', users=None, strict=True)`

Attempts to create a user with the provided identity and member users in the given service

#### Parameters

- **identity** (str) – Identity string for the user group
- **service** (str) – 'nifi' or 'registry'
- **users** (list) – A list of nifi.UserEntity or registry.User belonging to the group

- **strict** (*bool*) – Whether to throw an error on already exists

**Returns** The new (UserGroup) or (UserGroupEntity) object

`nipyapi.security.set_service_auth_token` (*token=None, token\_name='tokenAuth', service='nifi'*)

Helper method to set the auth token correctly for the specified service

**Parameters**

- **token** (*Optional[str]*) – The token to set. Defaults to None.
- **token\_name** (*str*) – the api\_key field name to set the token to. Defaults to 'tokenAuth'
- **service** (*str*) – 'nifi' or 'registry', the service to set

**Returns** True on success, False if token not set

**Return type** (*bool*)

`nipyapi.security.service_logout` (*service='nifi'*)

Logs out from the service by resetting the token :param service: 'nifi' or 'registry'; the target service :type service: str

**Returns** True of access removed, False if still set

**Return type** (*bool*)

`nipyapi.security.get_service_access_status` (*service='nifi', bool\_response=False*)

Gets the access status for the current session

**Parameters**

- **service** (*str*) – A String of 'nifi' or 'registry' to indicate which service to check status for
- **bool\_response** (*bool*) – If True, the function will return False on hitting an Error instead of raising it. Useful for connection testing.

**Returns** (*bool*) if bool\_response, else the Service Access Status of the User

`nipyapi.security.add_user_to_access_policy` (*user, policy, service='nifi', refresh=True, strict=True*)

Attempts to add the given user object to the given access policy

**Parameters**

- **user** (*User*) or (*UserEntity*) – User object to add
- **policy** (*AccessPolicyEntity*) or (*AccessPolicy*) – Access Policy object
- **service** (*str*) – 'nifi' or 'registry' to identify the target service
- **refresh** (*bool*) – Whether to refresh the policy object before submit
- **strict** (*bool*) – If True, will return error if user already present, if False will ignore the already exists

**Returns** Updated Policy object

`nipyapi.security.update_access_policy` (*policy, service='nifi'*)

Applies an updated access policy to the service indicated

**Parameters**

- **policy** (*PolicyEntity*) – The policy object to submit
- **service** (*str*) – 'nifi' or 'registry' to indicate the target service

**Returns** The updated policy if successful

**Return type** (PolicyEntity)

```
nipyapi.security.get_access_policy_for_resource(resource, action, r_id=None, service='nifi', auto_create=False)
```

Attempts to retrieve the access policy for a given resource and action, and optionally resource\_id if targeting NiFi. Optionally creates the policy if it doesn't already exist

**Parameters**

- **resource** (*str*) – A valid resource in the target service
- **action** (*str*) – A valid action, typically 'read', 'write' or 'delete'
- **r\_id** (*Optional[str]*) – The UUID of the resource, valid only if targeting NiFi resources
- **service** (*str*) – Which service to target, typically 'nifi' or 'registry'
- **auto\_create** (*bool*) – Whether to create the targeted policy if it doesn't already exist

**Returns** The relevant AccessPolicy object

```
nipyapi.security.create_access_policy(resource, action, r_id=None, service='nifi')
```

Creates an access policy for the given resource, action and optionally resource\_id for NiFi.

**Parameters**

- **resource** (*str*) – a valid resource type for this service, e.g. 'bucket'
- **action** (*str*) – a valid action type for this service, typically 'read', 'write' or 'delete'
- **r\_id** (*optional[str]*) – if NiFi, the resource ID of the resource
- **service** (*str*) – the service to target

**Returns** An access policy object for that service

```
nipyapi.security.list_service_users(service='nifi')
```

Lists all users of a given service, takes a service name as a string

```
nipyapi.security.get_service_user(identifier, identifier_type='identity', service='nifi')
```

Gets the unique user matching to the given identifier and type.

**Parameters**

- **identifier** (*str*) – the string to search for
- **identifier\_type** (*str*) – the field to search in
- **service** (*str*) – the name of the service

**Returns** None if no match, else single object

```
nipyapi.security.set_service_ssl_context(service='nifi',
                                         ca_file=None,
                                         client_cert_file=None, client_key_file=None,
                                         client_key_password=None,
                                         check_hostname=None)
```

Create an SSLContext for connecting over https to a secured NiFi or NiFi-Registry instance.

This method can be used to create an SSLContext for two-way TLS in which a client cert is used by the service to authenticate the client.

This method can also be used for one-way TLS in which the client verifies the server's certificate, but authenticates using a different form of credentials, such as LDAP username/password.

If you are using one-way TLS with a certificate signed by a root CA trusted by your system/platform, this step is not necessary as the default TLS-handshake should “just work.”

#### Parameters

- **service** (*str*) – ‘nifi’ or ‘registry’ to indicate which service config to set the ssl context to
- **ca\_file** (*str*) – A PEM file containing certs for the root CA(s) for the NiFi Registry server
- **client\_cert\_file** (*str*) – A PEM file containing the public certificates for the user/client identity
- **client\_key\_file** (*str*) – An encrypted (password-protected) PEM file containing the client’s secret key
- **client\_key\_password** (*str*) – The password to decrypt the client\_key\_file
- **check\_hostname** (*bool*) – Enable or Disable hostname checking

**Returns** (None)

`nipyapi.security.add_user_group_to_access_policy` (*user\_group*, *policy*, *service*=‘nifi’, *refresh*=True)

Attempts to add the given user group object to the given access policy

#### Parameters

- **user\_group** (*UserGroup*) or (*UserGroupEntity*) – User group object to add
- **policy** (*AccessPolicyEntity*) or (*AccessPolicy*) – Access Policy object
- **service** (*str*) – ‘nifi’ or ‘registry’ to identify the target service
- **refresh** (*bool*) – Whether to refresh the policy object before submission

**Returns** Updated Policy object

`nipyapi.security.bootstrap_security_policies` (*service*, *user\_identity*=None, *group\_identity*=None)

Creates a default security context within NiFi or NiFi-Registry

#### Parameters

- **service** (*str*) – ‘nifi’ or ‘registry’ to indicate which service
- **user\_identity** – a service user to establish in the security context
- **group\_identity** – a service group to establish in the security context

**Returns** None

`nipyapi.security.service_login` (*service*=‘nifi’, *username*=None, *password*=None, *bool\_response*=False)

Login to the currently configured NiFi or NiFi-Registry server.

Login requires a secure connection over https. Prior to calling this method, the host must be specified and the SSLContext should be configured (if necessary).

Successful login will result in a generated token (JWT) being cached in the api\_client config that will be passed in all future REST API calls. To clear that token, call service\_logout.

The token is temporary and will expire after a duration set by the server. After a token expires, you must call this method again to generate a new token.

#### Parameters

- **service** (*str*) – ‘nifi’ or ‘registry’; the service to login to
- **username** (*str*) – The username to submit
- **password** (*str*) – The password to use
- **bool\_response** (*bool*) – If True, the function will return False instead of an error. Useful for connection testing.

**Returns** True if successful, False or an Error if not. See bool\_response

**Return type** (bool)

`nipyapi.security.remove_service_user` (*user*, *service*=‘nifi’, *strict*=True)

Removes a given User from the given Service

**Parameters**

- **user** – [(nifi.UserEntity), (registry.User)] Target User object
- **service** (*str*) – ‘nifi’ or ‘registry’
- **strict** (*bool*) – Whether to throw an error if User not found

**Returns** Updated User Entity or None

`nipyapi.security.list_service_user_groups` (*service*=‘nifi’)

Returns list of service user groups for a given service :param service: ‘nifi’ or ‘registry’ :type service: str

**Returns** [(nifi.UserGroupEntity, registry.UserGroup)]

`nipyapi.security.get_service_user_group` (*identifier*, *identifier\_type*=‘identity’, *service*=‘nifi’)

Gets the unique group matching to the given identifier and type.

**Parameters**

- **identifier** (*str*) – the string to search for
- **identifier\_type** (*str*) – the field to search in, identity or id
- **service** (*str*) – the name of the service

**Returns** None if no match, else single object

`nipyapi.security.remove_service_user_group` (*group*, *service*=‘nifi’, *strict*=True)

Removes a given User Group from the given Service

**Parameters**

- **group** – [(nifi.UserEntity), (registry.User)] Target User object
- **service** (*str*) – ‘nifi’ or ‘registry’
- **strict** (*bool*) – Whether to throw an error if User not found

**Returns** Updated User Group or None

### 3.2.5 System

For system and cluster level functions interacting with the underlying NiFi Services

`nipyapi.system.get_system_diagnostics` ()

Returns NiFi Systems diagnostics page

Returns (json):

`nipyapi.system.get_cluster()`  
EXPERIMENTAL Returns the contents of the NiFi cluster

Returns (json):

`nipyapi.system.get_node(nid)`  
Returns the cluster node information

**Parameters** `nid` (*str*) – The UUID of the Node to target

Returns:

`nipyapi.system.get_nifi_version_info()`  
Returns the version information of the connected NiFi instance

Returns (VersionInfoDTO):

### 3.2.6 Templates

For Managing NiFi Templates

`nipyapi.templates.list_all_templates(native=True)`  
Gets a list of all templates on the canvas

**Returns** A list of TemplateEntity's

**Return type** (list[*TemplateEntity*])

`nipyapi.templates.get_template_by_name(name)`  
DEPRECATED Returns a specific template by name, if it exists.

Note: This function is replaced by `get_template`

**Parameters** `name` (*str*) – The Name of the template, exact match required

**Returns** (TemplateEntity)

`nipyapi.templates.deploy_template(pg_id, template_id, loc_x=0.0, loc_y=0.0)`  
Instantiates a given template request in a given process group

**Parameters**

- `pg_id` (*str*) – The UUID of the Process Group to deploy into
- `template_id` (*str*) – The UUID of the Template to deploy. Note that the Template must already be uploaded and available to the target Process Group
- `loc_x` (*float*) – The X coordinate to deploy the Template at. Default(0.0)
- `loc_y` (*float*) – The X coordinate to deploy the Template at. Default(0.0)

**Returns**

The FlowEntity of the Process Group with the deployed template

**Return type** (*FlowEntity*)

`nipyapi.templates.upload_template(pg_id, template_file)`  
Uploads a given template xml from from the file system to the given Process Group

**Parameters**

- `pg_id` (*str*) – The UUID of the Process Group to upload to
- `template_file` (*str*) – The path including filename to the template file

**Returns** The new Template object



**Return type** (*TemplateEntity*)

`nipyapi.templates.create_pg_snippet(pg_id)`

Creates a snippet of the targeted process group, and returns the object ready to be turned into a Template

**Parameters** `pg_id` – UUID of the process Group to snippet

**Returns** The Snippet Object

**Return type** (*SnippetEntity*)

`nipyapi.templates.create_template(pg_id, name, desc="")`

Creates a Template from a Process Group

**Parameters**

- `pg_id` (*str*) – The UUID of the target Process Group
- `name` (*str*) – The name for the new Template. Must be unique
- `desc` (*optional [str]*) – The description for the new Template

**Returns** The newly created Template

**Return type** (*TemplateEntity*)

`nipyapi.templates.delete_template(t_id)`

Deletes a Template

**Parameters** `t_id` (*str*) – UUID of the Template to be deleted

**Returns** The updated Template object

`nipyapi.templates.export_template(t_id, output='string', file_path=None)`

Exports a given Template as either a string or a file.

Note that to reimport the Template it must be a file

**Parameters**

- `t_id` (*str*) – The UUID of the Template to export
- `output` (*str*) – 'string' or 'file' to set the export action
- `file_path` (*Optional [str]*) – The full path including filename to write the Template export to

**Returns**

**A String representation of the exported Template XML. Note** that this may not be utf-8 encoded.

**Return type** (*str*)

`nipyapi.templates.get_template(identifier, identifier_type='name', greedy=False)`

Filters the list of all Templates for a given string in a given field. Note that filters are configured in config.py

**Parameters**

- `identifier` (*str*) – The string to filter on
- `identifier_type` (*str*) – The identifier of the field to filter on
- `greedy` (*bool*) – True for greedy match, False for exact match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.templates.load_template_from_xml_file_path(file_path)`

Loads a TemplateEntity from an xml file for a given path

**Parameters** `file_path` (*str*) – path to the xml file

**Returns** TemplateEntity

`nipyapi.templates.load_template_from_xml_file_stream(file_stream)`

Loads a TemplateEntity from a template xml file

**Parameters** `file_stream` (*io stream*) – the xml file stream as returned by open

**Returns** TemplateEntity

`nipyapi.templates.load_template_from_xml_string(xml_string)`

Loads a TemplateEntity from xml string, as if you had read in the xml file to string

**Parameters** `xml_string` (*str*) – string of xml

**Returns** TemplateEntity

### 3.2.7 Utils

Convenience utility functions for NiPyApi, not really intended for external use

`nipyapi.utils.dump(obj, mode='json')`

**Dumps a native datatype object or swagger entity to json or yaml** defaults to json

**Parameters**

- **obj** (*varies*) – The native datatype object or swagger type to serialise
- **mode** (*str*) – 'json' or 'yaml', the supported export modes

Returns (*str*): The serialised object

`nipyapi.utils.load(obj, dto=None)`

Loads a serialised object back into native datatypes, and optionally imports it back into the native NiFi DTO

Warning: Using this on objects not produced by this Package may have unintended results! While efforts have been made to ensure that unsafe loading is not possible, no stringent security testing has been completed.

**Parameters**

- **obj** (*dict, list*) – The serialised object to import
- **dto** (*Optional [tuple{str, str}]*) – A Tuple describing the service and
- **that should be constructed.** (*object*) –
- **dto = (e.g.)** –

**Returns:** Either the loaded object in native Python datatypes, or the constructed native datatype object

`nipyapi.utils.fs_read(file_path)`

Convenience function to read an Object from a FilePath

**Parameters** `file_path` (*str*) – The Full path including filename to read from

Returns: The object that was read

`nipyapi.utils.fs_write(obj, file_path)`

Convenience function to write an Object to a FilePath

**Parameters**

- **obj** (*varies*) – The Object to write out

- **file\_path** (*str*) – The Full path including filename to write to

Returns: The object that was written

`nipyapi.utils.filter_obj(obj, value, key, greedy=True)`

Implements a custom filter method because native datatypes don't have consistently named or located fields.

Note that each object used by this function must be registered with `identifier_types` and `identifiers` in `config`

#### Parameters

- **obj** (*varies*) – the NiFi or NiFi-Registry object to filter on
- **value** (*str*) – the String value to look for
- **key** (*str*) – the object key to filter against
- **greedy** (*bool*) – If True, the value will be matched anywhere in the string, if False it will require exact match

Returns: None if 0 matches, list if > 1, single Object entity if ==1

`nipyapi.utils.wait_to_complete(test_function, *args, **kwargs)`

Implements a basic return loop for a given function which is capable of a True/False output

#### Parameters

- **test\_function** – Function which returns a bool once the target state is reached
- **delay** (*int*) – The number of seconds between each attempt, defaults to `config.short_retry_delay`
- **max\_wait** (*int*) – the maximum number of seconds before issuing a Timeout, defaults to `config.short_max_wait`
- **\*args** – Any args to pass through to the test function
- **\*\*kwargs** – Any Keyword Args to pass through to the test function

Returns (bool): True for success, False for not

`nipyapi.utils.is_endpoint_up(endpoint_url)`

Tests if a URL is available for requests

**Parameters** **endpoint\_url** (*str*) – The URL to test

Returns (bool): True for a 200 response, False for not

`nipyapi.utils.set_endpoint(endpoint_url, ssl=False, login=False, username=None, password=None)`

#### EXPERIMENTAL

Sets the endpoint when switching between instances of NiFi or other projects. Not tested extensively with secured instances.

#### Parameters

- **endpoint\_url** (*str*) – The URL to set as the endpoint. Auto-detects the relevant service e.g. `'http://localhost:18080/nifi-registry-api'`
- **ssl** (*bool*) – Whether to use the default security context in `nipyapi.config` to authenticate if a secure URL is detected
- **login** (*bool*) – Whether to attempt login using default cred in `config` requires `ssl` to be set
- **username** (*str*) – The username to use for login, if specified
- **password** (*str*) – The password to use for login, if specified

Returns (bool): True for success, False for not

`nipyapi.utils.start_docker_containers(docker_containers, network_name='demo')`  
 Deploys a list of DockerContainer's on a given network

**Parameters**

- **docker\_containers** (*list [DockerContainer]*) – list of Dockers to start
- **network\_name** (*str*) – The name of the Docker Bridge Network to get or create for the Docker Containers

Returns: Nothing

**class** `nipyapi.utils.DockerContainer` (*name=None, image\_name=None, image\_tag=None, ports=None, env=None, volumes=None, test\_url=None, endpoint=None*)

Bases: object

Helper class for Docker container automation without using Ansible

**get\_container()**  
 Fetch the container object

**get\_test\_url\_status()**  
 Checks if a URL is available :return: status code if available, String 'ConnectionError' if not

**set\_container(container)**  
 Set the container object

`nipyapi.utils.infer_object_label_from_class(obj)`

**Returns the expected STRING label for an object class required by certain functions.**

**Parameters** `obj` – The object to infer the name of

**Returns** str of the relevant name, or raises an AssertionError

`nipyapi.utils.bypass_slash_encoding(service, bypass)`  
 Instructs the API Client to bypass encoding the '/' character

**Parameters**

- **service** (*str*) – 'nifi' or 'registry'
- **bypass** (*bool*) – True will not encode '/' in fields via API calls

**Returns** None

`nipyapi.utils.exception_handler(status_code=None, response=None)`  
 Simple Function wrapper to handle HTTP Status Exceptions

`nipyapi.utils.enforce_min_ver(min_version, bool_response=False, service='nifi')`  
 Raises an error if target NiFi environment is not minimum version :param min\_version: Version to check against  
 :type min\_version: str :param bool\_response: If True, will return True instead of raising error

**Returns** (bool) or (NotImplementedError)

`nipyapi.utils.check_version(base, comparator=None, service='nifi', default_version='0.2.0')`  
 Compares version base against either version comparator, or the version of the currently connected service instance.

Since NiFi is java, it may return a version with -SNAPSHOT as part of it. As such, that will be stripped from either the comparator version or the version returned from NiFi

#### Parameters

- **base** (*str*) – The base version for the comparison test
- **comparator** (*optional[str]*) – The version to compare against
- **default\_version** (*optional[str]*) – The version to assume the service is if the check cannot be completed
- **service** (*str*) – The service to test the version against, currently only supports NiFi

Returns (int): -1/0/1 if base is lower/equal/newer than comparator

`nipyapi.utils.validate_parameters_versioning_support` (*verify\_nifi=True, verify\_registry=True*)

Convenience method to check if Parameters are supported :param verify\_nifi: If True, check NiFi meets the min version :type verify\_nifi: bool :param verify\_registry: If True, check Registry meets the min version :type verify\_registry: bool

### 3.2.8 Versioning

For interactions with the NiFi Registry Service and related functions

`nipyapi.versioning.create_registry_client` (*name, uri, description*)

Creates a Registry Client in the NiFi Controller Services

#### Parameters

- **name** (*str*) – The name of the new Client
- **uri** (*str*) – The URI for the connection, such as ‘<http://registry:18080>’
- **description** (*str*) – A description for the Client

**Returns** The new registry client object

**Return type** (*RegistryClientEntity*)

`nipyapi.versioning.list_registry_clients` ()

Lists the available Registry Clients in the NiFi Controller Services

**Returns** (list[*RegistryClientEntity*]) objects

`nipyapi.versioning.delete_registry_client` (*client, refresh=True*)

Deletes a Registry Client from the list of NiFi Controller Services

#### Parameters

- **client** (*RegistryClientEntity*) – The client to delete
- **refresh** (*bool*) – Whether to refresh the object before action

**Returns** The updated client object

**Return type** (*RegistryClientEntity*)

`nipyapi.versioning.get_registry_client` (*identifier, identifier\_type='name'*)

Filters the Registry clients to a particular identifier

#### Parameters

- **identifier** (*str*) – the filter string

- **identifier\_type** (*str*) – the parameter to filter on

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.versioning.list_registry_buckets()`

Lists all available Buckets in the NiFi Registry

**Returns** (list[*Bucket*]) objects

`nipyapi.versioning.create_registry_bucket(name)`

Creates a new Registry Bucket

**Parameters** **name** (*str*) – name for the bucket, must be unique in the Registry

**Returns** The new Bucket object

**Return type** (*Bucket*)

`nipyapi.versioning.delete_registry_bucket(bucket)`

Removes a bucket from the NiFi Registry

**Parameters** **bucket** (*Bucket*) – the Bucket object to remove

**Returns** The updated Bucket object

**Return type** (*Bucket*)

`nipyapi.versioning.get_registry_bucket(identifier, identifier_type='name', greedy=True)`

Filters the Bucket list to a particular identifier

**Parameters**

- **identifier** (*str*) – the filter string
- **identifier\_type** (*str*) – the param to filter on
- **greedy** (*bool*) – False for exact match, True for greedy match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.versioning.save_flow_ver(process_group, registry_client, bucket, flow_name=None, flow_id=None, comment="", desc="", refresh=True, force=False)`

Adds a Process Group into NiFi Registry Version Control, or saves a new version to an existing VersionedFlow with a new version

**Parameters**

- **process\_group** (*ProcessGroupEntity*) – the ProcessGroup object to save as a new Flow Version
- **registry\_client** (*RegistryClient*) – The Client linked to the Registry which contains the Bucket to save to
- **bucket** (*Bucket*) – the Bucket on the NiFi Registry to save to
- **flow\_name** (*str*) – A name for the VersionedFlow in the Bucket Note you need either a name for a new VersionedFlow, or the ID of an existing one to save a new version
- **flow\_id** (*Optional [str]*) – Identifier of an existing VersionedFlow in the bucket, if saving a new version to an existing flow
- **comment** (*str*) – A comment for the version commit
- **desc** (*str*) – A description of the VersionedFlow
- **refresh** (*bool*) – Whether to refresh the object revisions before action

- **force** (*bool*) – Whether to Force Commit, or just regular Commit

**Returns** (VersionControlInformationEntity)

`nipyapi.versioning.list_flows_in_bucket(bucket_id)`

List of all Flows in a given NiFi Registry Bucket

**Parameters** **bucket\_id** (*str*) – The UUID of the Bucket to fetch from

**Returns** (list[VersionedFlow]) objects

`nipyapi.versioning.get_flow_in_bucket(bucket_id, identifier, identifier_type='name', greedy=True)`

Filters the Flows in a Bucket against a particular identifier

**Parameters**

- **bucket\_id** (*str*) – UUID of the Bucket to filter against
- **identifier** (*str*) – The string to filter on
- **identifier\_type** (*str*) – The param to check
- **greedy** (*bool*) – False for exact match, True for greedy match

**Returns** None for no matches, Single Object for unique match, list(Objects) for multiple matches

`nipyapi.versioning.get_latest_flow_ver(bucket_id, flow_id)`

Gets the most recent version of a VersionedFlowSnapshot from a bucket

**Parameters**

- **bucket\_id** (*str*) – the UUID of the Bucket containing the flow
- **flow\_id** (*str*) – the UUID of the VersionedFlow to be retrieved

**Returns** (VersionedFlowSnapshot)

`nipyapi.versioning.update_flow_ver(process_group, target_version=None)`

Changes a versioned flow to the specified version, or the latest version

**Parameters**

- **process\_group** (*ProcessGroupEntity*) – ProcessGroupEntity under version control to change
- **target\_version** (*Optional [None, Int]*) – Either None to move to the
- **available version, or Int of the version number to move to (latest)** –

**Returns** True if successful, False if not

**Return type** (bool)

`nipyapi.versioning.get_version_info(process_group)`

Gets the Version Control information for a particular Process Group

**Parameters** **process\_group** (*ProcessGroupEntity*) – the ProcessGroup to work with

**Returns** (VersionControlInformationEntity)

`nipyapi.versioning.create_flow(bucket_id, flow_name, flow_desc="", flow_type='Flow')`

Creates a new VersionedFlow stub in NiFi Registry. Can be used to write VersionedFlow information to without using a NiFi Process Group directly

**Parameters**

- **bucket\_id** (*str*) – UUID of the Bucket to write to

- **flow\_name** (*str*) – Name for the new VersionedFlow object
- **flow\_desc** (*Optional [str]*) – Description for the new VersionedFlow object
- **flow\_type** (*Optional [str]*) – Type of the VersionedFlow, should be ‘Flow’

**Returns** (VersionedFlow)

`nipyapi.versioning.create_flow_version(flow, flow_snapshot, refresh=True)`  
EXPERIMENTAL

Writes a FlowSnapshot into a VersionedFlow as a new version update

Note that this differs from `save_flow_ver` which creates a new Flow Version containing the snapshot. This function writes a snapshot to an existing Flow Version. Useful in migrating Flow Versions between environments.

#### Parameters

- **flow** (*VersionedFlowObject*) – the VersionedFlow object to write to
- **flow\_snapshot** (*VersionedFlowSnapshot*) – the Snapshot to write into the VersionedFlow
- **refresh** (*bool*) – Whether to refresh the object status before actioning

**Returns** The new (VersionedFlowSnapshot)

`nipyapi.versioning.get_flow_version(bucket_id, flow_id, version=None, export=False)`  
Retrieves the latest, or a specific, version of a Flow

#### Parameters

- **bucket\_id** (*str*) – the UUID of the bucket containing the Flow
- **flow\_id** (*str*) – the UUID of the Flow to be retrieved from the Bucket
- **version** (*Optional [None, str]*) – ‘None’ to retrieve the latest version, or a version number as a string to get that version
- **export** (*bool*) – True to get the raw json object from the server for export, False to get the native DataType

**Returns** If `export=False`, or the raw json otherwise

**Return type** (*VersionedFlowSnapshot*)

WARNING: This call is impacted by <https://issues.apache.org/jira/browse/NIFIREG-135> Which means you sometimes can’t trust the version count

`nipyapi.versioning.export_flow_version(bucket_id, flow_id, version=None, file_path=None, mode='json')`

Convenience method to export the identified VersionedFlowSnapshot in the provided format mode.

#### Parameters

- **bucket\_id** (*str*) – the UUID of the bucket containing the Flow
- **flow\_id** (*str*) – the UUID of the Flow to be retrieved from the Bucket
- **version** (*Optional [None, Str]*) – ‘None’ to retrieve the latest version, or a version number as a string to get that version
- **file\_path** (*str*) – The path and filename to write to. Defaults to None which returns the serialised obj
- **mode** (*str*) – ‘json’ or ‘yaml’ to specific the encoding format

**Returns** (str) of the encoded Snapshot



```
nipyapi.versioning.import_flow_version(bucket_id, encoded_flow=None, file_path=None,
                                       flow_name=None, flow_id=None)
```

Imports a given encoded\_flow version into the bucket and flow described, may optionally be passed a file to read the encoded flow\_contents from.

Note that only one of encoded\_flow or file\_path, and only one of flow\_name or flow\_id should be specified.

#### Parameters

- **bucket\_id** (*str*) – UUID of the bucket to write the encoded\_flow version
- **encoded\_flow** (*Optional [str]*) – The encoded flow to import; if not specified file\_path is read from.
- **file\_path** (*Optional [str]*) – The file path to read the encoded flow from , if not specified encoded\_flow is read from.
- **flow\_name** (*Optional [str]*) – If this is to be the first version in a new flow object, then this is the String name for the flow object.
- **flow\_id** (*Optional [str]*) – If this is a new version for an existing flow object, then this is the ID of that object.

**Returns** The new (VersionedFlowSnapshot)

```
nipyapi.versioning.list_flow_versions(bucket_id, flow_id, registry_id=None, service='registry')
```

EXPERIMENTAL List all the versions of a given Flow in a given Bucket

#### Parameters

- **bucket\_id** (*str*) – UUID of the bucket holding the flow to be enumerated
- **flow\_id** (*str*) – UUID of the flow in the bucket to be enumerated
- **registry\_id** (*str*) – UUID of the registry client linking the bucket, only required if requesting flows via NiFi instead of directly Registry
- **service** (*str*) – Accepts 'nifi' or 'registry', indicating which service to query

#### Returns

**list**(VersionedFlowSnapshotMetadata) or (VersionedFlowSnapshotMetadataSetEntity)

```
nipyapi.versioning.deploy_flow_version(parent_id, location, bucket_id, flow_id,
                                       reg_client_id, version=None)
```

Deploys a versioned flow as a new process group inside the given parent process group. If version is not provided, the latest version will be deployed.

#### Parameters

- **parent\_id** (*str*) – The ID of the parent Process Group to create the new process group in.
- **location** (*tuple[x, y]*) – the x,y coordinates to place the new Process Group under the parent
- **bucket\_id** (*str*) – ID of the bucket containing the versioned flow to deploy.
- **reg\_client\_id** (*str*) – ID of the registry client connection to use.
- **flow\_id** (*str*) – ID of the versioned flow to deploy.
- **version** (*Optional [int, str]*) – version to deploy, if not provided latest version will be deployed.

**Returns** (ProcessGroupEntity) of the newly deployed Process Group

## 3.3 Swagger Client SDKs

These sub-packages are full swagger clients to the NiFi and NiFi-Registry APIs and may be used directly, or wrapped into the NiPyApi SDK convenience functions

### 3.3.1 NiFi Swagger Client

#### 3.3.1.1 Subpackages

**nipyapi.nifi.apis package**

#### Submodules

**nipyapi.nifi.apis.access\_api module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.apis.access_api.AccessApi (api_client=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

```
create_access_token (**kwargs)
```

Creates a token for accessing the REST API via username/password The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. It is stored in the browser as a cookie, but also returned in the response body to be stored/used by third party client scripts. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **username** (*str*) –
- **password** (*str*) –

**Returns** str If the method is called asynchronously, returns the request thread.

```
create_access_token_from_ticket (**kwargs)
```

Creates a token for accessing the REST API via Kerberos ticket exchange / SPNEGO negotiation The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. It is also stored in the browser as a cookie. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when

receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_from\_ticket(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_from\_ticket\_with\_http\_info** (\*\*kwargs)

Creates a token for accessing the REST API via Kerberos ticket exchange / SPNEGO negotiation The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. It is also stored in the browser as a cookie. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_from\_ticket\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_with\_http\_info** (\*\*kwargs)

Creates a token for accessing the REST API via username/password The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. It is stored in the browser as a cookie, but also returned in the response body to be stored/used by third party client scripts. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_with\_http\_info(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **username** (*str*) –
- **password** (*str*) –

**Returns** str If the method is called asynchronously, returns the request thread.

**get\_access\_status** (\*\*kwargs)

Gets the status the client's access Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_status(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** AccessStatusEntity If the method is called asynchronously, returns the request thread.

**get\_access\_status\_with\_http\_info** (\*\*kwargs)

Gets the status the client's access Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_status\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `AccessStatusEntity` If the method is called asynchronously, returns the request thread.

**get\_access\_token\_expiration** (*\*\*kwargs*)

Get expiration for current Access Token Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_token\_expiration(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `AccessTokenExpirationEntity` If the method is called asynchronously, returns the request thread.

**get\_access\_token\_expiration\_with\_http\_info** (*\*\*kwargs*)

Get expiration for current Access Token Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_token\_expiration\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `AccessTokenExpirationEntity` If the method is called asynchronously, returns the request thread.

**get\_login\_config** (*\*\*kwargs*)

Retrieves the access configuration for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_login\_config(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `AccessConfigurationEntity` If the method is called asynchronously, returns the request thread.

**get\_login\_config\_with\_http\_info** (*\*\*kwargs*)

Retrieves the access configuration for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_login\_config\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `AccessConfigurationEntity` If the method is called asynchronously, returns the request thread.

**knox\_callback** (*\*\*kwargs*)

Redirect/callback URI for processing the result of the Apache Knox login sequence. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request

by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_callback(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**knox\_callback\_with\_http\_info** (\*\*kwargs)

Redirect/callback URI for processing the result of the Apache Knox login sequence. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_callback\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**knox\_logout** (\*\*kwargs)

Performs a logout in the Apache Knox. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_logout(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**knox\_logout\_with\_http\_info** (\*\*kwargs)

Performs a logout in the Apache Knox. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_logout\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**knox\_request** (\*\*kwargs)

Initiates a request to authenticate through Apache Knox. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_request(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**knox\_request\_with\_http\_info** (\*\*kwargs)

Initiates a request to authenticate through Apache Knox. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default.

To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.knox\_request\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**log\_out** (*\*\*kwargs*)

Performs a logout for other providers that have been issued a JWT. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**log\_out\_complete** (*\*\*kwargs*)

Completes the logout sequence by removing the cached Logout Request and Cookie if they existed and redirects to /nifi/login. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out\_complete(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**log\_out\_complete\_with\_http\_info** (*\*\*kwargs*)

Completes the logout sequence by removing the cached Logout Request and Cookie if they existed and redirects to /nifi/login. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out\_complete\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**log\_out\_with\_http\_info** (*\*\*kwargs*)

Performs a logout for other providers that have been issued a JWT. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.connections\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.connections\_api.ConnectionsApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**delete\_connection** (id, \*\*kwargs)

Deletes a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_connection(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ConnectionEntity If the method is called asynchronously, returns the request thread.

**delete\_connection\_with\_http\_info** (id, \*\*kwargs)

Deletes a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_connection\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ConnectionEntity If the method is called asynchronously, returns the request thread.

**get\_connection** (*id*, *\*\*kwargs*)

Gets a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connection(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** ConnectionEntity If the method is called asynchronously, returns the request thread.

**get\_connection\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connection\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** ConnectionEntity If the method is called asynchronously, returns the request thread.

**update\_connection** (*id*, *body*, *\*\*kwargs*)

Updates a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_connection(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **body** ([ConnectionEntity](#)) – The connection configuration details. (required)

**Returns** ConnectionEntity If the method is called asynchronously, returns the request thread.

**update\_connection\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_connection\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)



- **body** (*ConnectionEntity*) – The connection configuration details. (required)

**Returns** *ConnectionEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.controller\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.controller\_api.**ControllerApi** (*api\_client=None*)

Bases: *object*

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_bulletin** (*body, \*\*kwargs*)

Creates a new bulletin

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_bulletin(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*BulletinEntity*) – The reporting task configuration details. (required)

**Returns** *BulletinEntity* If the method is called asynchronously, returns the request thread.

**create\_bulletin\_with\_http\_info** (*body, \*\*kwargs*)

Creates a new bulletin

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_bulletin\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*BulletinEntity*) – The reporting task configuration details. (required)

**Returns** *BulletinEntity* If the method is called asynchronously, returns the request thread.

**create\_controller\_service** (*body, \*\*kwargs*)

Creates a new controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_controller\_service(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**create\_controller\_service\_with\_http\_info** (*body*, *\*\*kwargs*)

Creates a new controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_controller\_service\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**create\_registry\_client** (*body*, *\*\*kwargs*)

Creates a new registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_registry\_client(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*RegistryClientEntity*) – The registry configuration details. (required)

**Returns** *RegistryClientEntity* If the method is called asynchronously, returns the request thread.

**create\_registry\_client\_with\_http\_info** (*body*, *\*\*kwargs*)

Creates a new registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_registry\_client\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*RegistryClientEntity*) – The registry configuration details. (required)

**Returns** *RegistryClientEntity* If the method is called asynchronously, returns the request thread.

**create\_reporting\_task** (*body*, *\*\*kwargs*)

Creates a new reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_reporting\_task(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ReportingTaskEntity*) – The reporting task configuration details. (required)

**Returns** *ReportingTaskEntity* If the method is called asynchronously, returns the request thread.

**create\_reporting\_task\_with\_http\_info** (*body*, *\*\*kwargs*)

Creates a new reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_reporting\_task\_with\_http\_info(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ReportingTaskEntity*) – The reporting task configuration details. (required)

**Returns** *ReportingTaskEntity* If the method is called asynchronously, returns the request thread.

**delete\_history** (*end\_date*, *\*\*kwargs*)

Purges history

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_history(end\_date, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **end\_date** (*str*) – Purge actions before this date/time. (required)

**Returns** *HistoryEntity* If the method is called asynchronously, returns the request thread.

**delete\_history\_with\_http\_info** (*end\_date*, *\*\*kwargs*)

Purges history

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_history\_with\_http\_info(end\_date, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **end\_date** (*str*) – Purge actions before this date/time. (required)

**Returns** *HistoryEntity* If the method is called asynchronously, returns the request thread.

**delete\_node** (*id*, *\*\*kwargs*)

Removes a node from the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_node(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)

**Returns** NodeEntity If the method is called asynchronously, returns the request thread.

**delete\_node\_with\_http\_info** (*id*, *\*\*kwargs*)

Removes a node from the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_node\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)

**Returns** NodeEntity If the method is called asynchronously, returns the request thread.

**delete\_registry\_client** (*id*, *\*\*kwargs*)

Deletes a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_registry\_client(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** RegistryClientEntity If the method is called asynchronously, returns the request thread.

**delete\_registry\_client\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_registry\_client\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** RegistryClientEntity If the method is called asynchronously, returns the request thread.

**get\_cluster** (*\*\*kwargs*)

Gets the contents of the cluster Returns the contents of the cluster including all nodes and their status. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_cluster(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ClusterEntity If the method is called asynchronously, returns the request thread.

**get\_cluster\_with\_http\_info** (*\*\*kwargs*)

Gets the contents of the cluster Returns the contents of the cluster including all nodes and their status. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_cluster\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ClusterEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_config** (*\*\*kwargs*)

Retrieves the configuration for this NiFi Controller

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_config(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerConfigurationEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_config\_with\_http\_info** (*\*\*kwargs*)

Retrieves the configuration for this NiFi Controller

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_config\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerConfigurationEntity If the method is called asynchronously, returns the request thread.

**get\_node** (*id*, *\*\*kwargs*)

Gets a node in the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>>

```
def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_node(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)

**Returns** NodeEntity If the method is called asynchronously, returns the request thread.

**get\_node\_status\_history** (*\*\*kwargs*)

Gets status history for the node Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_node\_status\_history(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ComponentHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_node\_status\_history\_with\_http\_info** (*\*\*kwargs*)

Gets status history for the node Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_node\_status\_history\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ComponentHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_node\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a node in the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_node\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)

**Returns** NodeEntity If the method is called asynchronously, returns the request thread.

**get\_registry\_client** (*id*, *\*\*kwargs*)

Gets a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registry\_client(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The registry id. (required)

**Returns** RegistryClientEntity If the method is called asynchronously, returns the request thread.

**get\_registry\_client\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registry\_client\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)

**Returns** RegistryClientEntity If the method is called asynchronously, returns the request thread.

**get\_registry\_clients** (*\*\*kwargs*)

Gets the listing of available registry clients

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registry\_clients(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RegistryClientsEntity If the method is called asynchronously, returns the request thread.

**get\_registry\_clients\_with\_http\_info** (*\*\*kwargs*)

Gets the listing of available registry clients

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registry\_clients\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RegistryClientsEntity If the method is called asynchronously, returns the request thread.

**update\_controller\_config** (*body*, *\*\*kwargs*)

Retrieves the configuration for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_config(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ControllerConfigurationEntity*) – The controller configuration. (required)

**Returns** ControllerConfigurationEntity If the method is called asynchronously, returns the request thread.

**update\_controller\_config\_with\_http\_info** (*body*, *\*\*kwargs*)

Retrieves the configuration for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_config\_with\_http\_info(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ControllerConfigurationEntity*) – The controller configuration. (required)

**Returns** *ControllerConfigurationEntity* If the method is called asynchronously, returns the request thread.

**update\_node** (*id*, *body*, *\*\*kwargs*)

Updates a node in the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_node(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)
- **body** (*NodeEntity*) – The node configuration. The only configuration that will be honored at this endpoint is the status. (required)

**Returns** *NodeEntity* If the method is called asynchronously, returns the request thread.

**update\_node\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a node in the cluster

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_node\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The node id. (required)
- **body** (*NodeEntity*) – The node configuration. The only configuration that will be honored at this endpoint is the status. (required)

**Returns** *NodeEntity* If the method is called asynchronously, returns the request thread.

**update\_registry\_client** (*id*, *body*, *\*\*kwargs*)

Updates a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_registry\_client(id, body, callback=callback\_function)

**Parameters**



- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)
- **body** (*RegistryClientEntity*) – The registry configuration details. (required)

**Returns** *RegistryClientEntity* If the method is called asynchronously, returns the request thread.

**update\_registry\_client\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a registry client

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_registry\_client\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)
- **body** (*RegistryClientEntity*) – The registry configuration details. (required)

**Returns** *RegistryClientEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.controller\_services\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen>

**class** `nipyapi.nifi.apis.controller_services_api.ControllerServicesApi` (*api\_client=None*)  
Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**analyze\_configuration** (*id*, *body*, *\*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ConfigurationAnalysisEntity*) – The configuration analysis request. (required)

**Returns** *ConfigurationAnalysisEntity* If the method is called asynchronously, returns the request thread.

**analyze\_configuration\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ConfigurationAnalysisEntity*) – The configuration analysis request. (required)

**Returns** ConfigurationAnalysisEntity If the method is called asynchronously, returns the request thread.

**clear\_state** (*id*, *\*\*kwargs*)

Clears the state for a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**clear\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Clears the state for a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**delete\_validation\_request** (*id*, *request\_id*, *\*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def

```
callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_validation_request(id,
request_id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Controller Service (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_validation\_request\_with\_http\_info** (*id, request\_id, \*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_validation\_request\_with\_http\_info(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Controller Service (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_service** (*id, \*\*kwargs*)

Gets a controller service If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_service(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **ui\_only** (*bool*) –

**Returns** ControllerServiceEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_service\_references** (*id, \*\*kwargs*)

Gets a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_service\_references(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ControllerServiceReferencingComponentsEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_service\_references\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_service\_references\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ControllerServiceReferencingComponentsEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_service\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a controller service If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_service\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **ui\_only** (*bool*) –

**Returns** ControllerServiceEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor** (*id*, *property\_name*, *\*\*kwargs*)

Gets a controller service property descriptor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_property\_descriptor(id, property\_name, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **property\_name** (*str*) – The property name to return the descriptor for. (required)
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor\_with\_http\_info** (*id*, *property\_name*, *\*\*kwargs*)

Gets a controller service property descriptor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_property\_descriptor\_with\_http\_info(id, property\_name, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **property\_name** (*str*) – The property name to return the descriptor for. (required)
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_state** (*id*, *\*\*kwargs*)

Gets the state for a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the state for a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request** (*id*, *request\_id*, *\*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID. Once an Verification Request has been created, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request(id, request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Controller Service (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request\_with\_http\_info** (*id, request\_id, \*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID. Once an Verification Request has been created, that request can subsequently be retrieved via this end-point, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request\_with\_http\_info(id, request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Controller Service (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_controller\_service** (*id, \*\*kwargs*)

Deletes a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_controller\_service(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ControllerServiceEntity If the method is called asynchronously, returns the request thread.

**remove\_controller\_service\_with\_http\_info** (*id, \*\*kwargs*)

Deletes a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_controller\_service\_with\_http\_info(id, callback=callback\_function)

### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** `ControllerServiceEntity` If the method is called asynchronously, returns the request thread.

**submit\_config\_verification\_request** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Controller Service's configuration This will initiate the process of verifying a given Controller Service configuration. This may be a long-running task. As a result, this endpoint will immediately return a `ControllerServiceConfigVerificationRequestEntity`, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/controller-services/{serviceId}/verification-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/controller-services/{serviceId}/verification-requests/{requestId}`. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> thread = api.submit_config_verification_request(id, body, callback=callback_function)
```

### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*VerifyConfigRequestEntity*) – The controller service configuration verification request. (required)

**Returns** `VerifyConfigRequestEntity` If the method is called asynchronously, returns the request thread.

**submit\_config\_verification\_request\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Controller Service's configuration This will initiate the process of verifying a given Controller Service configuration. This may be a long-running task. As a result, this endpoint will immediately return a `ControllerServiceConfigVerificationRequestEntity`, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/controller-services/{serviceId}/verification-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/controller-services/{serviceId}/verification-requests/{requestId}`. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> thread = api.submit_config_verification_request_with_http_info(id, body, callback=callback_function)
```

### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)

- **body** (*VerifyConfigRequestEntity*) – The controller service configuration verification request. (required)

**Returns** *VerifyConfigRequestEntity* If the method is called asynchronously, returns the request thread.

**update\_controller\_service** (*id*, *body*, *\*\*kwargs*)

Updates a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_service(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**update\_controller\_service\_references** (*id*, *body*, *\*\*kwargs*)

Updates a controller services references

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_service\_references(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*UpdateControllerServiceReferenceRequestEntity*) – The controller service request update request. (required)

**Returns** *ControllerServiceReferencingComponentsEntity* If the method is called asynchronously, returns the request thread.

**update\_controller\_service\_references\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a controller services references

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_service\_references\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*UpdateControllerServiceReferenceRequestEntity*) – The controller service request update request. (required)

**Returns** *ControllerServiceReferencingComponentsEntity* If the method is called asynchronously, returns the request thread.



**update\_controller\_service\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_controller\_service\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status** (*id*, *body*, *\*\*kwargs*)

Updates run status of a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ControllerServiceRunStatusEntity*) – The controller service run status. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates run status of a controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The controller service id. (required)
- **body** (*ControllerServiceRunStatusEntity*) – The controller service run status. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.counters\_api module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.apis.counters_api.CountersApi (api_client=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

```
get_counters (**kwargs)
```

Gets the current counters for this NiFi Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_counters(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** CountersEntity If the method is called asynchronously, returns the request thread.

```
get_counters_with_http_info (**kwargs)
```

Gets the current counters for this NiFi Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_counters\_with\_http\_info(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** CountersEntity If the method is called asynchronously, returns the request thread.

```
update_counter (id, **kwargs)
```

Updates the specified counter. This will reset the counter value to 0 Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_counter(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the counter. (required)

**Returns** CounterEntity If the method is called asynchronously, returns the request thread.

**update\_counter\_with\_http\_info** (*id*, *\*\*kwargs*)

Updates the specified counter. This will reset the counter value to 0 Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_counter\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the counter. (required)

**Returns** CounterEntity If the method is called asynchronously, returns the request thread.

### nipyapi.nifi.apis.data\_transfer\_api module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.data\_transfer\_api.**DataTransferApi** (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually. Ref: <https://github.com/swagger-api/swagger-codegen>

**commit\_input\_port\_transaction** (*response\_code*, *port\_id*, *transaction\_id*, *\*\*kwargs*)

Commit or cancel the specified transaction

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.commit\_input\_port\_transaction(response\_code, port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **response\_code** (*int*) – The response code. Available values are BAD\_CHECKSUM(19), CONFIRM\_TRANSACTION(12) or CANCEL\_TRANSACTION(15). (required)
- **port\_id** (*str*) – The input port id. (required)
- **transaction\_id** (*str*) – The transaction id. (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

**commit\_input\_port\_transaction\_with\_http\_info** (*response\_code*, *port\_id*, *transaction\_id*, *\*\*kwargs*)

Commit or cancel the specified transaction

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread =

```
api.commit_input_port_transaction_with_http_info(response_code, port_id, transaction_id, call-  
back=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **response\_code** (*int*) – The response code. Available values are BAD\_CHECKSUM(19), CONFIRM\_TRANSACTION(12) or CANCEL\_TRANSACTION(15). (required)
- **port\_id** (*str*) – The input port id. (required)
- **transaction\_id** (*str*) – The transaction id. (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
commit_output_port_transaction(response_code, checksum, port_id, transaction_id,  
                               **kwargs)
```

Commit or cancel the specified transaction

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.commit\_output\_port\_transaction(response\_code, checksum, port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **response\_code** (*int*) – The response code. Available values are CONFIRM\_TRANSACTION(12) or CANCEL\_TRANSACTION(15). (required)
- **checksum** (*str*) – A checksum calculated at client side using CRC32 to check flow file content integrity. It must match with the value calculated at server side. (required)
- **port\_id** (*str*) – The output port id. (required)
- **transaction\_id** (*str*) – The transaction id. (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
commit_output_port_transaction_with_http_info(response_code, checksum, port_id,  
                                              transaction_id, **kwargs)
```

Commit or cancel the specified transaction

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.commit\_output\_port\_transaction\_with\_http\_info(response\_code, checksum, port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **response\_code** (*int*) – The response code. Available values are CONFIRM\_TRANSACTION(12) or CANCEL\_TRANSACTION(15). (required)
- **checksum** (*str*) – A checksum calculated at client side using CRC32 to check flow file content integrity. It must match with the value calculated at server side. (required)

- **port\_id** (*str*) – The output port id. (required)
- **transaction\_id** (*str*) – The transaction id. (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

**create\_port\_transaction** (*port\_type*, *port\_id*, *\*\*kwargs*)

Create a transaction to the specified output port or input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_port\_transaction(port\_type, port\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_type** (*str*) – The port type. (required)
- **port\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

**create\_port\_transaction\_with\_http\_info** (*port\_type*, *port\_id*, *\*\*kwargs*)

Create a transaction to the specified output port or input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_port\_transaction\_with\_http\_info(port\_type, port\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_type** (*str*) – The port type. (required)
- **port\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

**extend\_input\_port\_transaction\_ttl** (*port\_id*, *transaction\_id*, *\*\*kwargs*)

Extend transaction TTL

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.extend\_input\_port\_transaction\_ttl(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – (required)
- **transaction\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
extend_input_port_transaction_ttl_with_http_info (port_id, transaction_id,  
                                                  **kwargs)
```

Extend transaction TTL

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.extend\_input\_port\_transaction\_ttl\_with\_http\_info(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – (required)
- **transaction\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
extend_output_port_transaction_ttl (port_id, transaction_id, **kwargs)
```

Extend transaction TTL

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.extend\_output\_port\_transaction\_ttl(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – (required)
- **transaction\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
extend_output_port_transaction_ttl_with_http_info (port_id, transaction_id,  
                                                  **kwargs)
```

Extend transaction TTL

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.extend\_output\_port\_transaction\_ttl\_with\_http\_info(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – (required)
- **transaction\_id** (*str*) – (required)

**Returns** TransactionResultEntity If the method is called asynchronously, returns the request thread.

```
receive_flow_files (port_id, transaction_id, **kwargs)
```

Transfer flow files to the input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.receive\_flow\_files(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – The input port id. (required)
- **transaction\_id** (*str*) – (required)

**Returns** *str* If the method is called asynchronously, returns the request thread.

**receive\_flow\_files\_with\_http\_info** (*port\_id, transaction\_id, \*\*kwargs*)

Transfer flow files to the input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.receive\_flow\_files\_with\_http\_info(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – The input port id. (required)
- **transaction\_id** (*str*) – (required)

**Returns** *str* If the method is called asynchronously, returns the request thread.

**transfer\_flow\_files** (*port\_id, transaction\_id, \*\*kwargs*)

Transfer flow files from the output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.transfer\_flow\_files(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – The output port id. (required)
- **transaction\_id** (*str*) – (required)

**Returns** *StreamingOutput* If the method is called asynchronously, returns the request thread.

**transfer\_flow\_files\_with\_http\_info** (*port\_id, transaction\_id, \*\*kwargs*)

Transfer flow files from the output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.transfer\_flow\_files\_with\_http\_info(port\_id, transaction\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **port\_id** (*str*) – The output port id. (required)
- **transaction\_id** (*str*) – (required)

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.flow\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.flow\_api.**FlowApi** (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**activate\_controller\_services** (*id, body, \*\*kwargs*)

Enable or disable Controller Services in the specified Process Group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.activate\_controller\_services(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ActivateControllerServicesEntity*) – The request to schedule or unschedule. If the components in the request are not specified, all authorized components will be considered. (required)

**Returns** ActivateControllerServicesEntity If the method is called asynchronously, returns the request thread.

**activate\_controller\_services\_with\_http\_info** (*id, body, \*\*kwargs*)

Enable or disable Controller Services in the specified Process Group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.activate\_controller\_services\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ActivateControllerServicesEntity*) – The request to schedule or unschedule. If the components in the request are not specified, all authorized components will be considered. (required)

**Returns** ActivateControllerServicesEntity If the method is called asynchronously, returns the request thread.



**generate\_client\_id** (\*\*kwargs)

Generates a client id.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.generate\_client\_id(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**generate\_client\_id\_with\_http\_info** (\*\*kwargs)

Generates a client id.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.generate\_client\_id\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**get\_about\_info** (\*\*kwargs)

Retrieves details about this NiFi to put in the About dialog

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_about\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** AboutEntity If the method is called asynchronously, returns the request thread.

**get\_about\_info\_with\_http\_info** (\*\*kwargs)

Retrieves details about this NiFi to put in the About dialog

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_about\_info\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** AboutEntity If the method is called asynchronously, returns the request thread.

**get\_action** (id, \*\*kwargs)

Gets an action Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_action(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The action id. (required)

**Returns** ActionEntity If the method is called asynchronously, returns the request thread.

**get\_action\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets an action Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_action\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The action id. (required)

**Returns** ActionEntity If the method is called asynchronously, returns the request thread.

**get\_banners** (*\*\*kwargs*)

Retrieves the banners for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_banners(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** BannerEntity If the method is called asynchronously, returns the request thread.

**get\_banners\_with\_http\_info** (*\*\*kwargs*)

Retrieves the banners for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_banners\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** BannerEntity If the method is called asynchronously, returns the request thread.

**get\_buckets** (*id*, *\*\*kwargs*)

Gets the buckets from the specified registry for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_buckets(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)

**Returns** BucketsEntity If the method is called asynchronously, returns the request thread.

**get\_buckets\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the buckets from the specified registry for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def call-

```
back_function(response): >>> pprint(response) >>> >>> thread = api.get_buckets_with_http_info(id,
callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The registry id. (required)

**Returns** BucketsEntity If the method is called asynchronously, returns the request thread.

**get\_bulletin\_board** (*\*\*kwargs*)

Gets current bulletins

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bulletin\_board(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **after** (*str*) – Includes bulletins with an id after this value.
- **source\_name** (*str*) – Includes bulletins originating from this sources whose name match this regular expression.
- **message** (*str*) – Includes bulletins whose message that match this regular expression.
- **source\_id** (*str*) – Includes bulletins originating from this sources whose id match this regular expression.
- **group\_id** (*str*) – Includes bulletins originating from this sources whose group id match this regular expression.
- **limit** (*str*) – The number of bulletins to limit the response to.

**Returns** BulletinBoardEntity If the method is called asynchronously, returns the request thread.

**get\_bulletin\_board\_with\_http\_info** (*\*\*kwargs*)

Gets current bulletins

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bulletin\_board\_with\_http\_info(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **after** (*str*) – Includes bulletins with an id after this value.
- **source\_name** (*str*) – Includes bulletins originating from this sources whose name match this regular expression.
- **message** (*str*) – Includes bulletins whose message that match this regular expression.
- **source\_id** (*str*) – Includes bulletins originating from this sources whose id match this regular expression.
- **group\_id** (*str*) – Includes bulletins originating from this sources whose group id match this regular expression.
- **limit** (*str*) – The number of bulletins to limit the response to.

**Returns** BulletinBoardEntity If the method is called asynchronously, returns the request thread.

**get\_bulletins** (\*\*kwargs)

Retrieves Controller level bulletins

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bulletins(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerBulletinsEntity If the method is called asynchronously, returns the request thread.

**get\_bulletins\_with\_http\_info** (\*\*kwargs)

Retrieves Controller level bulletins

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bulletins\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerBulletinsEntity If the method is called asynchronously, returns the request thread.

**get\_cluster\_summary** (\*\*kwargs)

The cluster summary for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_cluster\_summary(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ClusteSummaryEntity If the method is called asynchronously, returns the request thread.

**get\_cluster\_summary\_with\_http\_info** (\*\*kwargs)

The cluster summary for this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_cluster\_summary\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ClusteSummaryEntity If the method is called asynchronously, returns the request thread.

**get\_component\_history** (*component\_id*, \*\*kwargs)

Gets configuration history for a component Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To

make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_component_history(component_id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **component\_id** (*str*) – The component id. (required)

**Returns** ComponentHistoryEntity If the method is called asynchronously, returns the request thread.

#### **get\_component\_history\_with\_http\_info** (*component\_id*, *\*\*kwargs*)

Gets configuration history for a component Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_component_history_with_http_info(component_id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **component\_id** (*str*) – The component id. (required)

**Returns** ComponentHistoryEntity If the method is called asynchronously, returns the request thread.

#### **get\_connection\_statistics** (*id*, *\*\*kwargs*)

Gets statistics for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_connection_statistics(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the statistics.

**Returns** ConnectionStatisticsEntity If the method is called asynchronously, returns the request thread.

#### **get\_connection\_statistics\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets statistics for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_connection_statistics_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the statistics.

**Returns** ConnectionStatisticsEntity If the method is called asynchronously, returns the request thread.

**get\_connection\_status** (*id*, *\*\*kwargs*)

Gets status for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connection\_status(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ConnectionStatusEntity If the method is called asynchronously, returns the request thread.

**get\_connection\_status\_history** (*id*, *\*\*kwargs*)

Gets the status history for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connection\_status\_history(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_connection\_status\_history\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the status history for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connection\_status\_history\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_connection\_status\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets status for a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.get_connection_status_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ConnectionStatusEntity If the method is called asynchronously, returns the request thread.

#### **get\_controller\_service\_types** (*\*\*kwargs*)

Retrieves the types of controller services that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.get_controller_service_types(callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **service\_type** (*str*) – If specified, will only return controller services that are compatible with this type of service.
- **service\_bundle\_group** (*str*) – If serviceType specified, is the bundle group of the serviceType.
- **service\_bundle\_artifact** (*str*) – If serviceType specified, is the bundle artifact of the serviceType.
- **service\_bundle\_version** (*str*) – If serviceType specified, is the bundle version of the serviceType.
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.
- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type\_filter** (*str*) – If specified, will only return types whose fully qualified class-name matches.

**Returns** ControllerServiceTypesEntity If the method is called asynchronously, returns the request thread.

#### **get\_controller\_service\_types\_with\_http\_info** (*\*\*kwargs*)

Retrieves the types of controller services that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.get_controller_service_types_with_http_info(callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **service\_type** (*str*) – If specified, will only return controller services that are compatible with this type of service.
- **service\_bundle\_group** (*str*) – If serviceType specified, is the bundle group of the serviceType.
- **service\_bundle\_artifact** (*str*) – If serviceType specified, is the bundle artifact of the serviceType.
- **service\_bundle\_version** (*str*) – If serviceType specified, is the bundle version of the serviceType.
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.
- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type\_filter** (*str*) – If specified, will only return types whose fully qualified class-name matches.

**Returns** ControllerServiceTypesEntity If the method is called asynchronously, returns the request thread.

#### **get\_controller\_services\_from\_controller** (*\*\*kwargs*)

Gets controller services for reporting tasks If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_services\_from\_controller(callback=callback\_function)

##### **Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **ui\_only** (*bool*) –

**Returns** ControllerServicesEntity If the method is called asynchronously, returns the request thread.

#### **get\_controller\_services\_from\_controller\_with\_http\_info** (*\*\*kwargs*)

Gets controller services for reporting tasks If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_services\_from\_controller\_with\_http\_info(callback=callback\_function)

##### **Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **ui\_only** (*bool*) –

**Returns** ControllerServicesEntity If the method is called asynchronously, returns the request thread.



**get\_controller\_services\_from\_group** (*id*, *\*\*kwargs*)

Gets all controller services. If the `uiOnly` query parameter is provided with a value of `true`, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_controller_services_from_group(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_ancestor\_groups** (*bool*) – Whether or not to include parent/ancestry process groups
- **include\_descendant\_groups** (*bool*) – Whether or not to include descendant process groups
- **ui\_only** (*bool*) –

**Returns** `ControllerServicesEntity` If the method is called asynchronously, returns the request thread.

**get\_controller\_services\_from\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all controller services. If the `uiOnly` query parameter is provided with a value of `true`, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_controller_services_from_group_with_http_info(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_ancestor\_groups** (*bool*) – Whether or not to include parent/ancestry process groups
- **include\_descendant\_groups** (*bool*) – Whether or not to include descendant process groups
- **ui\_only** (*bool*) –

**Returns** `ControllerServicesEntity` If the method is called asynchronously, returns the request thread.

**get\_controller\_status** (*\*\*kwargs*)

Gets the current status of this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_controller_status(callback=callback_function)
```

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerStatusEntity If the method is called asynchronously, returns the request thread.

**get\_controller\_status\_with\_http\_info** (\*\*kwargs)

Gets the current status of this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_controller\_status\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerStatusEntity If the method is called asynchronously, returns the request thread.

**get\_current\_user** (\*\*kwargs)

Retrieves the user identity of the user making the request

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_current\_user(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** CurrentUserEntity If the method is called asynchronously, returns the request thread.

**get\_current\_user\_with\_http\_info** (\*\*kwargs)

Retrieves the user identity of the user making the request

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_current\_user\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** CurrentUserEntity If the method is called asynchronously, returns the request thread.

**get\_flow** (*id*, \*\*kwargs)

Gets a process group If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **ui\_only** (*bool*) –

**Returns** ProcessGroupFlowEntity If the method is called asynchronously, returns the request thread.

**get\_flow\_config** (*\*\*kwargs*)

Retrieves the configuration for this NiFi flow

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_flow_config(callback=callback_function)
```

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** FlowConfigurationEntity If the method is called asynchronously, returns the request thread.

**get\_flow\_config\_with\_http\_info** (*\*\*kwargs*)

Retrieves the configuration for this NiFi flow

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_flow_config_with_http_info(callback=callback_function)
```

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** FlowConfigurationEntity If the method is called asynchronously, returns the request thread.

**get\_flow\_metrics** (*producer, \*\*kwargs*)

Gets all metrics for the flow from a particular node

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_flow_metrics(producer, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **producer** (*str*) – The producer for flow file metrics. Each producer may have its own output format. (required)
- **included\_registries** (*list[str]*) – Set of included metrics registries
- **sample\_name** (*str*) – Regular Expression Pattern to be applied against the sample name field
- **sample\_label\_value** (*str*) – Regular Expression Pattern to be applied against the sample label value field
- **root\_field\_name** (*str*) – Name of the first field of JSON object. Applicable for JSON producer only.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_flow\_metrics\_with\_http\_info** (*producer, \*\*kwargs*)

Gets all metrics for the flow from a particular node

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_flow_metrics_with_http_info(producer, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **producer** (*str*) – The producer for flow file metrics. Each producer may have its own output format. (required)
- **included\_registries** (*list[str]*) – Set of included metrics registries
- **sample\_name** (*str*) – Regular Expression Pattern to be applied against the sample name field
- **sample\_label\_value** (*str*) – Regular Expression Pattern to be applied against the sample label value field
- **root\_field\_name** (*str*) – Name of the first field of JSON object. Applicable for JSON producer only.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_flow\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a process group If the uiOnly query parameter is provided with a value of true, the returned entity may only contain fields that are necessary for rendering the NiFi User Interface. As such, the selected fields may change at any time, even during incremental releases, without warning. As a result, this parameter should not be provided by any client other than the UI. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **ui\_only** (*bool*) –

**Returns** ProcessGroupFlowEntity If the method is called asynchronously, returns the request thread.

**get\_flows** (*registry\_id*, *bucket\_id*, *\*\*kwargs*)

Gets the flows from the specified registry and bucket for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flows(registry\_id, bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **registry\_id** (*str*) – The registry id. (required)
- **bucket\_id** (*str*) – The bucket id. (required)

**Returns** VersionedFlowsEntity If the method is called asynchronously, returns the request thread.

**get\_flows\_with\_http\_info** (*registry\_id*, *bucket\_id*, *\*\*kwargs*)

Gets the flows from the specified registry and bucket for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the re-

```
sponse. >>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.get_flows_with_http_info(registry_id, bucket_id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **registry\_id** (*str*) – The registry id. (required)
- **bucket\_id** (*str*) – The bucket id. (required)

**Returns** VersionedFlowsEntity If the method is called asynchronously, returns the request thread.

**get\_input\_port\_status** (*id*, *\*\*kwargs*)

Gets status for an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_port\_status(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** PortStatusEntity If the method is called asynchronously, returns the request thread.

**get\_input\_port\_status\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets status for an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_port\_status\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** PortStatusEntity If the method is called asynchronously, returns the request thread.

**get\_output\_port\_status** (*id*, *\*\*kwargs*)

Gets status for an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_port\_status(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** PortStatusEntity If the method is called asynchronously, returns the request thread.

**get\_output\_port\_status\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets status for an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_port\_status\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** PortStatusEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_contexts** (*\*\*kwargs*)

Gets all Parameter Contexts

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_parameter\_contexts(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ParameterContextsEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_contexts\_with\_http\_info** (*\*\*kwargs*)

Gets all Parameter Contexts

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_parameter\_contexts\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ParameterContextsEntity If the method is called asynchronously, returns the request thread.

**get\_prioritizers** (*\*\*kwargs*)

Retrieves the types of prioritizers that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default.

To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_prioritizers(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** PrioritizerTypesEntity If the method is called asynchronously, returns the request thread.

**get\_prioritizers\_with\_http\_info** (*\*\*kwargs*)

Retrieves the types of prioritizers that this NiFi supports Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_prioritizers_with_http_info(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** PrioritizerTypesEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group\_status** (*id*, *\*\*kwargs*)

Gets the status for a process group The status for a process group includes status for all descendent components. When invoked on the root group with recursive set to true, it will return the current status of every component in the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_process_group_status(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **recursive** (*bool*) – Whether all descendant groups and the status of their content will be included. Optional, defaults to false
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ProcessGroupStatusEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group\_status\_history** (*id*, *\*\*kwargs*)

Gets status history for a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_process_group_status_history(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group\_status\_history\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets status history for a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_group\_status\_history\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group\_status\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the status for a process group The status for a process group includes status for all descendent components. When invoked on the root group with recursive set to true, it will return the current status of every component in the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_group\_status\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **recursive** (*bool*) – Whether all descendant groups and the status of their content will be included. Optional, defaults to false
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ProcessGroupStatusEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_status** (*id*, *\*\*kwargs*)

Gets status for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_status(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ProcessorStatusEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_status\_history** (*id*, *\*\*kwargs*)

Gets status history for a processor



This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_status\_history(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

#### `get_processor_status_history_with_http_info` (*id*, *\*\*kwargs*)

Gets status history for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_status\_history\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

#### `get_processor_status_with_http_info` (*id*, *\*\*kwargs*)

Gets status for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_status\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** ProcessorStatusEntity If the method is called asynchronously, returns the request thread.

#### `get_processor_types` (*\*\*kwargs*)

Retrieves the types of processors that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_types(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.

- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type** (*str*) – If specified, will only return types whose fully qualified classname matches.

**Returns** ProcessorTypesEntity If the method is called asynchronously, returns the request thread.

#### **get\_processor\_types\_with\_http\_info** (\*\*kwargs)

Retrieves the types of processors that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_types\_with\_http\_info(callback=callback\_function)

##### **Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.
- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type** (*str*) – If specified, will only return types whose fully qualified classname matches.

**Returns** ProcessorTypesEntity If the method is called asynchronously, returns the request thread.

#### **get\_registries** (\*\*kwargs)

Gets the listing of available registries

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registries(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RegistryClientsEntity If the method is called asynchronously, returns the request thread.

#### **get\_registries\_with\_http\_info** (\*\*kwargs)

Gets the listing of available registries

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_registries\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RegistryClientsEntity If the method is called asynchronously, returns the request thread.

#### **get\_remote\_process\_group\_status** (*id*, \*\*kwargs)

Gets status for a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_remote\_process\_group\_status(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** RemoteProcessGroupStatusEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_group\_status\_history** (*id*, *\*\*kwargs*)

Gets the status history

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_remote_process_group_status_history(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_group\_status\_history\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the status history

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_remote_process_group_status_history_with_http_info(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)

**Returns** StatusHistoryEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_group\_status\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets status for a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_remote_process_group_status_with_http_info(id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** RemoteProcessGroupStatusEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_task\_types** (*\*\*kwargs*)

Retrieves the types of reporting tasks that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_reporting\_task\_types(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.
- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type** (*str*) – If specified, will only return types whose fully qualified classname matches.

**Returns** ReportingTaskTypesEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_task\_types\_with\_http\_info** (*\*\*kwargs*)

Retrieves the types of reporting tasks that this NiFi supports Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_reporting\_task\_types\_with\_http\_info(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bundle\_group\_filter** (*str*) – If specified, will only return types that are a member of this bundle group.
- **bundle\_artifact\_filter** (*str*) – If specified, will only return types that are a member of this bundle artifact.
- **type** (*str*) – If specified, will only return types whose fully qualified classname matches.

**Returns** ReportingTaskTypesEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_tasks** (*\*\*kwargs*)

Gets all reporting tasks

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_reporting\_tasks(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ReportingTasksEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_tasks\_with\_http\_info** (*\*\*kwargs*)

Gets all reporting tasks

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_reporting\_tasks\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ReportingTasksEntity If the method is called asynchronously, returns the request thread.

**get\_runtime\_manifest** (\*\*kwargs)

Retrieves the runtime manifest for this NiFi instance. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_runtime\_manifest(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RuntimeManifestEntity If the method is called asynchronously, returns the request thread.

**get\_runtime\_manifest\_with\_http\_info** (\*\*kwargs)

Retrieves the runtime manifest for this NiFi instance. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_runtime\_manifest\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** RuntimeManifestEntity If the method is called asynchronously, returns the request thread.

**get\_templates** (\*\*kwargs)

Gets all templates

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_templates(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** TemplatesEntity If the method is called asynchronously, returns the request thread.

**get\_templates\_with\_http\_info** (\*\*kwargs)

Gets all templates

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_templates\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** TemplatesEntity If the method is called asynchronously, returns the request thread.

**get\_versions** (*registry\_id*, *bucket\_id*, *flow\_id*, *\*\*kwargs*)

Gets the flow versions from the specified registry and bucket for the specified flow for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_versions(registry\_id, bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **registry\_id** (*str*) – The registry id. (required)
- **bucket\_id** (*str*) – The bucket id. (required)
- **flow\_id** (*str*) – The flow id. (required)

**Returns** VersionedFlowSnapshotMetadataSetEntity If the method is called asynchronously, returns the request thread.

**get\_versions\_with\_http\_info** (*registry\_id*, *bucket\_id*, *flow\_id*, *\*\*kwargs*)

Gets the flow versions from the specified registry and bucket for the specified flow for the current user

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_versions\_with\_http\_info(registry\_id, bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **registry\_id** (*str*) – The registry id. (required)
- **bucket\_id** (*str*) – The bucket id. (required)
- **flow\_id** (*str*) – The flow id. (required)

**Returns** VersionedFlowSnapshotMetadataSetEntity If the method is called asynchronously, returns the request thread.

**query\_history** (*offset*, *count*, *\*\*kwargs*)

Gets configuration history Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.query\_history(offset, count, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **offset** (*str*) – The offset into the result set. (required)
- **count** (*str*) – The number of actions to return. (required)
- **sort\_column** (*str*) – The field to sort on.
- **sort\_order** (*str*) – The direction to sort.
- **start\_date** (*str*) – Include actions after this date.
- **end\_date** (*str*) – Include actions before this date.
- **user\_identity** (*str*) – Include actions performed by this user.

- **source\_id** (*str*) – Include actions on this component.

**Returns** HistoryEntity If the method is called asynchronously, returns the request thread.

**query\_history\_with\_http\_info** (*offset, count, \*\*kwargs*)

Gets configuration history Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.query\_history\_with\_http\_info(offset, count, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **offset** (*str*) – The offset into the result set. (required)
- **count** (*str*) – The number of actions to return. (required)
- **sort\_column** (*str*) – The field to sort on.
- **sort\_order** (*str*) – The direction to sort.
- **start\_date** (*str*) – Include actions after this date.
- **end\_date** (*str*) – Include actions before this date.
- **user\_identity** (*str*) – Include actions performed by this user.
- **source\_id** (*str*) – Include actions on this component.

**Returns** HistoryEntity If the method is called asynchronously, returns the request thread.

**schedule\_components** (*id, body, \*\*kwargs*)

Schedule or unschedule components in the specified Process Group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.schedule\_components(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ScheduleComponentsEntity*) – The request to schedule or unschedule. If the components in the request are not specified, all authorized components will be considered. (required)

**Returns** ScheduleComponentsEntity If the method is called asynchronously, returns the request thread.

**schedule\_components\_with\_http\_info** (*id, body, \*\*kwargs*)

Schedule or unschedule components in the specified Process Group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.schedule\_components\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The process group id. (required)
- **body** (*ScheduleComponentsEntity*) – The request to schedule or unschedule. If the components in the request are not specified, all authorized components will be considered. (required)

**Returns** *ScheduleComponentsEntity* If the method is called asynchronously, returns the request thread.

**search\_cluster** (*q*, *\*\*kwargs*)

Searches the cluster for a node with the specified address Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.search\_cluster(q, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) – Node address to search for. (required)

**Returns** *ClusterSearchResultsEntity* If the method is called asynchronously, returns the request thread.

**search\_cluster\_with\_http\_info** (*q*, *\*\*kwargs*)

Searches the cluster for a node with the specified address Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.search\_cluster\_with\_http\_info(q, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) – Node address to search for. (required)

**Returns** *ClusterSearchResultsEntity* If the method is called asynchronously, returns the request thread.

**search\_flow** (*\*\*kwargs*)

Performs a search against this NiFi using the specified search term Only search results from authorized components will be returned. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.search\_flow(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) –
- **a** (*str*) –

**Returns** *SearchResultsEntity* If the method is called asynchronously, returns the request thread.

**search\_flow\_with\_http\_info** (*\*\*kwargs*)

Performs a search against this NiFi using the specified search term Only search results from authorized components will be returned. This method makes a synchronous HTTP request by default.



To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.search_flow_with_http_info(callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) –
- **a** (*str*) –

**Returns** SearchResultsEntity If the method is called asynchronously, returns the request thread.

### nipyapi.nifi.apis.flowfile\_queues\_api module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.flowfile\_queues\_api.**FlowfileQueuesApi** (*api\_client=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_drop\_request** (*id*, *\*\*kwargs*)

Creates a request to drop the contents of the queue in this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.create_drop_request(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**create\_drop\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Creates a request to drop the contents of the queue in this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.create_drop_request_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**create\_flow\_file\_listing** (*id*, *\*\*kwargs*)

Lists the contents of the queue in this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow\_file\_listing(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**create\_flow\_file\_listing\_with\_http\_info** (*id*, *\*\*kwargs*)

Lists the contents of the queue in this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow\_file\_listing\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_listing\_request** (*id*, *listing\_request\_id*, *\*\*kwargs*)

Cancels and/or removes a request to list the contents of this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_listing\_request(id, listing\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **listing\_request\_id** (*str*) – The listing request id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_listing\_request\_with\_http\_info** (*id*, *listing\_request\_id*, *\*\*kwargs*)

Cancels and/or removes a request to list the contents of this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_listing\_request\_with\_http\_info(id, listing\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **listing\_request\_id** (*str*) – The listing request id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**download\_flow\_file\_content** (*id*, *flowfile\_uuid*, *\*\*kwargs*)

Gets the content for a FlowFile in a Connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.download\_flow\_file\_content(id, flowfile\_uuid, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **flowfile\_uuid** (*str*) – The flowfile uuid. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**download\_flow\_file\_content\_with\_http\_info** (*id*, *flowfile\_uuid*, *\*\*kwargs*)

Gets the content for a FlowFile in a Connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.download\_flow\_file\_content\_with\_http\_info(id, flowfile\_uuid, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **flowfile\_uuid** (*str*) – The flowfile uuid. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_drop\_request** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Gets the current status of a drop request for the specified connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_drop\_request(id, drop\_request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**get\_drop\_request\_with\_http\_info** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Gets the current status of a drop request for the specified connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_drop\_request\_with\_http\_info(id, drop\_request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**get\_flow\_file** (*id*, *flowfile\_uuid*, *\*\*kwargs*)

Gets a FlowFile from a Connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_file(id, flowfile\_uuid, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **flowfile\_uuid** (*str*) – The flowfile uuid. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** FlowFileEntity If the method is called asynchronously, returns the request thread.

**get\_flow\_file\_with\_http\_info** (*id*, *flowfile\_uuid*, *\*\*kwargs*)

Gets a FlowFile from a Connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_file\_with\_http\_info(id, flowfile\_uuid, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **flowfile\_uuid** (*str*) – The flowfile uuid. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** FlowFileEntity If the method is called asynchronously, returns the request thread.

**get\_listing\_request** (*id*, *listing\_request\_id*, *\*\*kwargs*)

Gets the current status of a listing request for the specified connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_listing\_request(id, listing\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **listing\_request\_id** (*str*) – The listing request id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**get\_listing\_request\_with\_http\_info** (*id*, *listing\_request\_id*, *\*\*kwargs*)

Gets the current status of a listing request for the specified connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_listing\_request\_with\_http\_info(id, listing\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **listing\_request\_id** (*str*) – The listing request id. (required)

**Returns** ListingRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_drop\_request** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Cancels and/or removes a request to drop the contents of this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_drop\_request(id, drop\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_drop\_request\_with\_http\_info** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Cancels and/or removes a request to drop the contents of this connection.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_drop\_request\_with\_http\_info(id, drop\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The connection id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.funnel\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.funnel\_api.FunnelApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_funnel** (*id*, *\*\*kwargs*)

Gets a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_funnel(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)

**Returns** FunnelEntity If the method is called asynchronously, returns the request thread.

**get\_funnel\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_funnel\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)

**Returns** FunnelEntity If the method is called asynchronously, returns the request thread.

**remove\_funnel** (*id*, *\*\*kwargs*)

Deletes a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_funnel(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.

- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** `FunnelEntity` If the method is called asynchronously, returns the request thread.

**remove\_funnel\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_funnel\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** `FunnelEntity` If the method is called asynchronously, returns the request thread.

**update\_funnel** (*id*, *body*, *\*\*kwargs*)

Updates a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_funnel(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)
- **body** (`FunnelEntity`) – The funnel configuration details. (required)

**Returns** `FunnelEntity` If the method is called asynchronously, returns the request thread.

**update\_funnel\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_funnel\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The funnel id. (required)
- **body** (`FunnelEntity`) – The funnel configuration details. (required)

**Returns** FunnelEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.input\_ports\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.input\_ports\_api.InputPortsApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_input\_port** (id, \*\*kwargs)

Gets an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_port(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**get\_input\_port\_with\_http\_info** (id, \*\*kwargs)

Gets an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_port\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**remove\_input\_port** (id, \*\*kwargs)

Deletes an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_input\_port(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)



- **id** (*str*) – The input port id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**remove\_input\_port\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.remove_input_port_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**update\_input\_port** (*id*, *body*, *\*\*kwargs*)

Updates an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_input_port(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)
- **body** (PortEntity) – The input port configuration details. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**update\_input\_port\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_input_port_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The input port id. (required)
- **body** (*PortEntity*) – The input port configuration details. (required)

**Returns** *PortEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status** (*id*, *body*, *\*\*kwargs*)

Updates run status of an input-port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The port id. (required)
- **body** (*PortRunStatusEntity*) – The port run status. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates run status of an input-port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The port id. (required)
- **body** (*PortRunStatusEntity*) – The port run status. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.labels\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen>

**class** nipyapi.nifi.apis.labels\_api.LabelsApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_label** (*id*, *\*\*kwargs*)

Gets a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_label(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The label id. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**get\_label\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_label\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The label id. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**remove\_label** (*id*, *\*\*kwargs*)

Deletes a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_label(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The label id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**remove\_label\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_label\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The label id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**update\_label** (*id*, *body*, *\*\*kwargs*)

Updates a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_label(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The label id. (required)
- **body** (LabelEntity) – The label configuration details. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**update\_label\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_label\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The label id. (required)
- **body** (LabelEntity) – The label configuration details. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.output\_ports\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.apis.output_ports_api.OutputPortsApi (api_client=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
 Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_output\_port** (*id*, *\*\*kwargs*)

Gets an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_port(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**get\_output\_port\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_port\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**remove\_output\_port** (*id*, *\*\*kwargs*)

Deletes an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_output\_port(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**remove\_output\_port\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.remove_output_port_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**update\_output\_port** (*id, body, \*\*kwargs*)

Updates an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_output_port(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **body** (PortEntity) – The output port configuration details. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**update\_output\_port\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_output_port_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The output port id. (required)
- **body** (PortEntity) – The output port configuration details. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**update\_run\_status** (*id, body, \*\*kwargs*)

Updates run status of an output-port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_run_status(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The port id. (required)
- **body** (*PortRunStatusEntity*) – The port run status. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**update\_run\_status\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates run status of an output-port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The port id. (required)
- **body** (*PortRunStatusEntity*) – The port run status. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**nipyapi.nifi.apis.parameter\_contexts\_api module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.parameter\_contexts\_api.**ParameterContextsApi** (*api\_client=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_parameter\_context** (*body*, *\*\*kwargs*)  
Create a Parameter Context

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_parameter\_context(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ParameterContextEntity*) – The Parameter Context. (required)

**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**create\_parameter\_context\_with\_http\_info** (*body*, *\*\*kwargs*)

Create a Parameter Context

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_parameter\_context\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ParameterContextEntity*) – The Parameter Context. (required)

**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**delete\_parameter\_context** (*id*, *\*\*kwargs*)

Deletes the Parameter Context with the given ID Deletes the Parameter Context with the given ID. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_parameter\_context(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The Parameter Context ID. (required)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, a new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**delete\_parameter\_context\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes the Parameter Context with the given ID Deletes the Parameter Context with the given ID. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_parameter\_context\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The Parameter Context ID. (required)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, a new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.



**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**delete\_update\_request** (*context\_id*, *request\_id*, *\*\*kwargs*)

Deletes the Update Request with the given ID Deletes the Update Request with the given ID. After a request is created via a POST to /nifi-api/parameter-contexts/update-requests, it is expected that the client will properly clean up the request by DELETE'ing it, once the Update process has completed. If the request is deleted before the request completes, then the Update request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_update\_request(context\_id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the ParameterContext (required)
- **request\_id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_update\_request\_with\_http\_info** (*context\_id*, *request\_id*, *\*\*kwargs*)

Deletes the Update Request with the given ID Deletes the Update Request with the given ID. After a request is created via a POST to /nifi-api/parameter-contexts/update-requests, it is expected that the client will properly clean up the request by DELETE'ing it, once the Update process has completed. If the request is deleted before the request completes, then the Update request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_update\_request\_with\_http\_info(context\_id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the ParameterContext (required)
- **request\_id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_validation\_request** (*context\_id*, *id*, *\*\*kwargs*)

Deletes the Validation Request with the given ID Deletes the Validation Request with the given ID. After a request is created via a POST to /nifi-api/validation-contexts, it is expected that the client will properly clean up the request by DELETE'ing it, once the validation process has completed. If the request is deleted before the request completes, then the Validation request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_validation\_request(context\_id, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ParameterContextValidationRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_validation\_request\_with\_http\_info** (*context\_id, id, \*\*kwargs*)

Deletes the Validation Request with the given ID Deletes the Validation Request with the given ID. After a request is created via a POST to /nifi-api/validation-contexts, it is expected that the client will properly clean up the request by DELETE'ing it, once the validation process has completed. If the request is deleted before the request completes, then the Validation request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_validation\_request\_with\_http\_info(context\_id, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ParameterContextValidationRequestEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_context** (*id, \*\*kwargs*)

Returns the Parameter Context with the given ID Returns the Parameter Context with the given ID. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_parameter\_context(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Parameter Context (required)
- **include\_inherited\_parameters** (*bool*) – Whether or not to include inherited parameters from other parameter contexts, and therefore also overridden values. If true, the result will be the 'effective' parameter context.

**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_context\_update** (*context\_id, request\_id, \*\*kwargs*)

Returns the Update Request with the given ID Returns the Update Request with the given ID. Once an Update Request has been created by performing a POST to /nifi-api/parameter-contexts, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to

be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> thread = api.get\_parameter\_context\_update(context\_id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **request\_id** (*str*) – The ID of the Update Request (required)

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_context\_update\_with\_http\_info** (*context\_id, request\_id, \*\*kwargs*)

Returns the Update Request with the given ID Returns the Update Request with the given ID. Once an Update Request has been created by performing a POST to /nifi-api/parameter-contexts, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_parameter\_context\_update\_with\_http\_info(context\_id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **request\_id** (*str*) – The ID of the Update Request (required)

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**get\_parameter\_context\_with\_http\_info** (*id, \*\*kwargs*)

Returns the Parameter Context with the given ID Returns the Parameter Context with the given ID. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_parameter\_context\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Parameter Context (required)
- **include\_inherited\_parameters** (*bool*) – Whether or not to include inherited parameters from other parameter contexts, and therefore also overridden values. If true, the result will be the ‘effective’ parameter context.

**Returns** ParameterContextEntity If the method is called asynchronously, returns the request thread.

**get\_validation\_request** (*context\_id, id, \*\*kwargs*)

Returns the Validation Request with the given ID Returns the Validation Request with the given ID. Once a Validation Request has been created by performing a POST to /nifi-api/validation-contexts, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_validation\_request(context\_id, id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **id** (*str*) – The ID of the Validation Request (required)

**Returns** ParameterContextValidationRequestEntity If the method is called asynchronously, returns the request thread.

**get\_validation\_request\_with\_http\_info** (*context\_id, id, \*\*kwargs*)

Returns the Validation Request with the given ID Returns the Validation Request with the given ID. Once a Validation Request has been created by performing a POST to /nifi-api/validation-contexts, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_validation\_request\_with\_http\_info(context\_id, id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – The ID of the Parameter Context (required)
- **id** (*str*) – The ID of the Validation Request (required)

**Returns** ParameterContextValidationRequestEntity If the method is called asynchronously, returns the request thread.

**submit\_parameter\_context\_update** (*context\_id, body, \*\*kwargs*)

Initiate the Update Request of a Parameter Context This will initiate the process of updating a Parameter Context. Changing the value of a Parameter may require that one or more components be stopped and restarted, so this action may take significantly more time than many other REST API actions. As a result, this endpoint will immediately return a ParameterContextUpdateRequestEntity, and the process of updating the necessary components will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /parameter-contexts/update-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /parameter-contexts/update-requests/{requestId}. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_parameter\_context\_update(context\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – (required)
- **body** (*ParameterContextEntity*) – The updated version of the parameter context. (required)

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**submit\_parameter\_context\_update\_with\_http\_info** (*context\_id, body, \*\*kwargs*)

Initiate the Update Request of a Parameter Context This will initiate the process of updating a Parameter Context. Changing the value of a Parameter may require that one or more components be stopped and restarted, so this action may take significantly more time than many other REST API actions. As a result, this endpoint will immediately return a ParameterContextUpdateRequestEntity, and the process

of updating the necessary components will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/parameter-contexts/update-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/parameter-contexts/update-requests/{requestId}`. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_parameter\_context\_update\_with\_http\_info(context\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – (required)
- **body** (*ParameterContextEntity*) – The updated version of the parameter context. (required)

**Returns** ParameterContextUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### **submit\_validation\_request** (*context\_id, body, \*\*kwargs*)

Initiate a Validation Request to determine how the validity of components will change if a Parameter Context were to be updated This will initiate the process of validating all components whose Process Group is bound to the specified Parameter Context. Performing validation against an arbitrary number of components may be expect and take significantly more time than many other REST API actions. As a result, this endpoint will immediately return a ParameterContextValidationRequestEntity, and the process of validating the necessary components will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/parameter-contexts/validation-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/parameter-contexts/validation-requests/{requestId}`. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_validation\_request(context\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – (required)
- **body** (*ParameterContextValidationRequestEntity*) – The validation request (required)

**Returns** ParameterContextValidationRequestEntity If the method is called asynchronously, returns the request thread.

#### **submit\_validation\_request\_with\_http\_info** (*context\_id, body, \*\*kwargs*)

Initiate a Validation Request to determine how the validity of components will change if a Parameter Context were to be updated This will initiate the process of validating all components whose Process Group is bound to the specified Parameter Context. Performing validation against an arbitrary number of components may be expect and take significantly more time than many other REST API actions. As a result, this endpoint will immediately return a ParameterContextValidationRequestEntity, and the process of validating the necessary components will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/parameter-contexts/validation-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/parameter-contexts/validation-requests/{requestId}`. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_validation\_request\_with\_http\_info(context\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **context\_id** (*str*) – (required)
- **body** (*ParameterContextValidationRequestEntity*) – The validation request (required)

**Returns** *ParameterContextValidationRequestEntity* If the method is called asynchronously, returns the request thread.

**update\_parameter\_context** (*id, body, \*\*kwargs*)

Modifies a Parameter Context This endpoint will update a Parameter Context to match the provided entity. However, this request will fail if any component is running and is referencing a Parameter in the Parameter Context. Generally, this endpoint is not called directly. Instead, an update request should be submitted by making a POST to the `/parameter-contexts/update-requests` endpoint. That endpoint will, in turn, call this endpoint. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_parameter_context(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – (required)
- **body** (*ParameterContextEntity*) – The updated Parameter Context (required)

**Returns** *ParameterContextEntity* If the method is called asynchronously, returns the request thread.

**update\_parameter\_context\_with\_http\_info** (*id, body, \*\*kwargs*)

Modifies a Parameter Context This endpoint will update a Parameter Context to match the provided entity. However, this request will fail if any component is running and is referencing a Parameter in the Parameter Context. Generally, this endpoint is not called directly. Instead, an update request should be submitted by making a POST to the `/parameter-contexts/update-requests` endpoint. That endpoint will, in turn, call this endpoint. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_parameter_context_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – (required)
- **body** (*ParameterContextEntity*) – The updated Parameter Context (required)

**Returns** *ParameterContextEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.policies\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.apis.policies_api.PoliciesApi` (*api\_client=None*)

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_access\_policy** (*body, \*\*kwargs*)

Creates an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_policy(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*AccessPolicyEntity*) – The access policy configuration details. (required)

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**create\_access\_policy\_with\_http\_info** (*body, \*\*kwargs*)

Creates an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_policy\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*AccessPolicyEntity*) – The access policy configuration details. (required)

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**get\_access\_policy** (*id, \*\*kwargs*)

Gets an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**get\_access\_policy\_for\_resource** (*action, resource, \*\*kwargs*)

Gets an access policy for the specified action and resource Will return the effective policy if no component specific policy exists for the specified action and resource. Must have Read permissions to the policy with the desired action and resource. Permissions for the policy that is returned will be indicated in the response. This means the client could be authorized to get the policy for a given component but the effective policy may be inherited from an ancestor Process Group. If the client does not have permissions to that policy, the response will not include the policy and the permissions in the response

will be marked accordingly. If the client does not have permissions to the policy of the desired action and resource a 403 response will be returned. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_for\_resource(action, resource, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **action** (*str*) – The request action. (required)
- **resource** (*str*) – The resource of the policy. (required)

**Returns** AccessPolicyEntity If the method is called asynchronously, returns the request thread.

#### **get\_access\_policy\_for\_resource\_with\_http\_info** (*action, resource, \*\*kwargs*)

Gets an access policy for the specified action and resource Will return the effective policy if no component specific policy exists for the specified action and resource. Must have Read permissions to the policy with the desired action and resource. Permissions for the policy that is returned will be indicated in the response. This means the client could be authorized to get the policy for a given component but the effective policy may be inherited from an ancestor Process Group. If the client does not have permissions to that policy, the response will not include the policy and the permissions in the response will be marked accordingly. If the client does not have permissions to the policy of the desired action and resource a 403 response will be returned. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_for\_resource\_with\_http\_info(action, resource, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **action** (*str*) – The request action. (required)
- **resource** (*str*) – The resource of the policy. (required)

**Returns** AccessPolicyEntity If the method is called asynchronously, returns the request thread.

#### **get\_access\_policy\_with\_http\_info** (*id, \*\*kwargs*)

Gets an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)

**Returns** AccessPolicyEntity If the method is called asynchronously, returns the request thread.

#### **remove\_access\_policy** (*id, \*\*kwargs*)

Deletes an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_access\_policy(id, callback=callback\_function)

#### Parameters



- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**remove\_access\_policy\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes an access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.remove_access_policy_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**update\_access\_policy** (*id*, *body*, *\*\*kwargs*)

Updates a access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_access_policy(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **body** (`AccessPolicyEntity`) – The access policy configuration details. (required)

**Returns** `AccessPolicyEntity` If the method is called asynchronously, returns the request thread.

**update\_access\_policy\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_access_policy_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **body** (*AccessPolicyEntity*) – The access policy configuration details. (required)

**Returns** *AccessPolicyEntity* If the method is called asynchronously, returns the request thread.

**nipyapi.nifi.apis.process\_groups\_api module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` (*api\_client=None*)

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**copy\_snippet** (*id, body, \*\*kwargs*)

Copies a snippet and discards it.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.copy\_snippet(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*CopySnippetRequestEntity*) – The copy snippet request. (required)

**Returns** *FlowEntity* If the method is called asynchronously, returns the request thread.

**copy\_snippet\_with\_http\_info** (*id, body, \*\*kwargs*)

Copies a snippet and discards it.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.copy\_snippet\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*CopySnippetRequestEntity*) – The copy snippet request. (required)

**Returns** *FlowEntity* If the method is called asynchronously, returns the request thread.

**create\_connection** (*id*, *body*, *\*\*kwargs*)

Creates a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_connection(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ConnectionEntity*) – The connection configuration details. (required)

**Returns** *ConnectionEntity* If the method is called asynchronously, returns the request thread.

**create\_connection\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a connection

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_connection\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ConnectionEntity*) – The connection configuration details. (required)

**Returns** *ConnectionEntity* If the method is called asynchronously, returns the request thread.

**create\_controller\_service** (*id*, *body*, *\*\*kwargs*)

Creates a new controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_controller\_service(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**create\_controller\_service\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a new controller service

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_controller\_service\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ControllerServiceEntity*) – The controller service configuration details. (required)

**Returns** *ControllerServiceEntity* If the method is called asynchronously, returns the request thread.

**create\_empty\_all\_connections\_request** (*id*, *\*\*kwargs*)

Creates a request to drop all flowfiles of all connection queues in this process group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_empty\_all\_connections\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** *DropRequestEntity* If the method is called asynchronously, returns the request thread.

**create\_empty\_all\_connections\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Creates a request to drop all flowfiles of all connection queues in this process group.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_empty\_all\_connections\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** *DropRequestEntity* If the method is called asynchronously, returns the request thread.

**create\_funnel** (*id*, *body*, *\*\*kwargs*)

Creates a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_funnel(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*FunnelEntity*) – The funnel configuration details. (required)

**Returns** *FunnelEntity* If the method is called asynchronously, returns the request thread.

**create\_funnel\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a funnel

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_funnel\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*FunnelEntity*) – The funnel configuration details. (required)

**Returns** *FunnelEntity* If the method is called asynchronously, returns the request thread.

**create\_input\_port** (*id*, *body*, *\*\*kwargs*)

Creates an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_input\_port(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*PortEntity*) – The input port configuration details. (required)

**Returns** *PortEntity* If the method is called asynchronously, returns the request thread.

**create\_input\_port\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates an input port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_input\_port\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*PortEntity*) – The input port configuration details. (required)

**Returns** *PortEntity* If the method is called asynchronously, returns the request thread.

**create\_label** (*id*, *body*, *\*\*kwargs*)

Creates a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_label(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*LabelEntity*) – The label configuration details. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**create\_label\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a label

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_label\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (LabelEntity) – The label configuration details. (required)

**Returns** LabelEntity If the method is called asynchronously, returns the request thread.

**create\_output\_port** (*id*, *body*, *\*\*kwargs*)

Creates an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_output\_port(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (PortEntity) – The output port configuration. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**create\_output\_port\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates an output port

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_output\_port\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (PortEntity) – The output port configuration. (required)

**Returns** PortEntity If the method is called asynchronously, returns the request thread.

**create\_process\_group** (*id*, *body*, *\*\*kwargs*)

Creates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_process\_group(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupEntity*) – The process group configuration details. (required)

**Returns** *ProcessGroupEntity* If the method is called asynchronously, returns the request thread.

**create\_process\_group\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_process\_group\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupEntity*) – The process group configuration details. (required)

**Returns** *ProcessGroupEntity* If the method is called asynchronously, returns the request thread.

**create\_processor** (*id*, *body*, *\*\*kwargs*)

Creates a new processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_processor(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessorEntity*) – The processor configuration details. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**create\_processor\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a new processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_processor\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessorEntity*) – The processor configuration details. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**create\_remote\_process\_group** (*id*, *body*, *\*\*kwargs*)

Creates a new process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_remote\_process\_group(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*RemoteProcessGroupEntity*) – The remote process group configuration details. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**create\_remote\_process\_group\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a new process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_remote\_process\_group\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*RemoteProcessGroupEntity*) – The remote process group configuration details. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**create\_template** (*id*, *body*, *\*\*kwargs*)

Creates a template and discards the specified snippet.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_template(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*CreateTemplateRequestEntity*) – The create template request. (required)

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

**create\_template\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Creates a template and discards the specified snippet.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_template\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters



- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*CreateTemplateRequestEntity*) – The create template request. (required)

**Returns** *TemplateEntity* If the method is called asynchronously, returns the request thread.

#### **delete\_replace\_process\_group\_request** (*id*, *\*\*kwargs*)

Deletes the Replace Request with the given ID Deletes the Replace Request with the given ID. After a request is created via a POST to `/process-groups/{id}/replace-requests`, it is expected that the client will properly clean up the request by DELETE'ing it, once the Replace process has completed. If the request is deleted before the request completes, then the Replace request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_replace_process_group_request(id, callback=callback_function)
```

##### **Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** *ProcessGroupReplaceRequestEntity* If the method is called asynchronously, returns the request thread.

#### **delete\_replace\_process\_group\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes the Replace Request with the given ID Deletes the Replace Request with the given ID. After a request is created via a POST to `/process-groups/{id}/replace-requests`, it is expected that the client will properly clean up the request by DELETE'ing it, once the Replace process has completed. If the request is deleted before the request completes, then the Replace request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_replace_process_group_request_with_http_info(id, callback=callback_function)
```

##### **Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** *ProcessGroupReplaceRequestEntity* If the method is called asynchronously, returns the request thread.

#### **delete\_variable\_registry\_update\_request** (*group\_id*, *update\_id*, *\*\*kwargs*)

Deletes an update request for a process group's variable registry. If the request is not yet complete, it will automatically be cancelled. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_variable_registry_update_request(group_id, update_id, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **group\_id** (*str*) – The process group id. (required)
- **update\_id** (*str*) – The ID of the Variable Registry Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VariableRegistryUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_variable\_registry\_update\_request\_with\_http\_info** (*group\_id*, *update\_id*,  
\*\**kwargs*)

Deletes an update request for a process group's variable registry. If the request is not yet complete, it will automatically be cancelled. Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_variable\_registry\_update\_request\_with\_http\_info(group\_id, update\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **group\_id** (*str*) – The process group id. (required)
- **update\_id** (*str*) – The ID of the Variable Registry Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VariableRegistryUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**export\_process\_group** (*id*, \*\**kwargs*)

Gets a process group for download

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_process\_group(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_referenced\_services** (*bool*) – If referenced services from outside the target group should be included

**Returns** str If the method is called asynchronously, returns the request thread.

**export\_process\_group\_with\_http\_info** (*id*, \*\**kwargs*)

Gets a process group for download

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_process\_group\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_referenced\_services** (*bool*) – If referenced services from outside the target group should be included

**Returns** *str* If the method is called asynchronously, returns the request thread.

**get\_connections** (*id*, *\*\*kwargs*)

Gets all connections

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connections(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** *ConnectionsEntity* If the method is called asynchronously, returns the request thread.

**get\_connections\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all connections

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_connections\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** *ConnectionsEntity* If the method is called asynchronously, returns the request thread.

**get\_drop\_all\_flowfiles\_request** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Gets the current status of a drop all flowfiles request.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_drop\_all\_flowfiles\_request(id, drop\_request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** *DropRequestEntity* If the method is called asynchronously, returns the request thread.

**get\_drop\_all\_flowfiles\_request\_with\_http\_info** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Gets the current status of a drop all flowfiles request.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_drop\_all\_flowfiles\_request\_with\_http\_info(id, drop\_request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**get\_funnels** (*id*, *\*\*kwargs*)

Gets all funnels

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_funnels(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** FunnelsEntity If the method is called asynchronously, returns the request thread.

**get\_funnels\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all funnels

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_funnels\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** FunnelsEntity If the method is called asynchronously, returns the request thread.

**get\_input\_ports** (*id*, *\*\*kwargs*)

Gets all input ports

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_ports(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** InputPortsEntity If the method is called asynchronously, returns the request thread.

**get\_input\_ports\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all input ports

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_ports\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** InputPortsEntity If the method is called asynchronously, returns the request thread.

**get\_labels** (*id*, *\*\*kwargs*)

Gets all labels

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_labels(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** LabelsEntity If the method is called asynchronously, returns the request thread.

**get\_labels\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all labels

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_labels\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** LabelsEntity If the method is called asynchronously, returns the request thread.

**get\_local\_modifications** (*id*, *\*\*kwargs*)

Gets a list of local modifications to the Process Group since it was last synchronized with the Flow Registry

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_local\_modifications(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** FlowComparisonEntity If the method is called asynchronously, returns the request thread.

**get\_local\_modifications\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a list of local modifications to the Process Group since it was last synchronized with the Flow Registry

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_local\_modifications\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** FlowComparisonEntity If the method is called asynchronously, returns the request thread.

**get\_output\_ports** (*id*, *\*\*kwargs*)

Gets all output ports

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_ports(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** OutputPortsEntity If the method is called asynchronously, returns the request thread.

**get\_output\_ports\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all output ports

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_ports\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** OutputPortsEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group** (*id*, *\*\*kwargs*)

Gets a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_group(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**get\_process\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_group\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

#### **get\_process\_groups** (*id*, *\*\*kwargs*)

Gets all process groups

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_groups(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupsEntity If the method is called asynchronously, returns the request thread.

#### **get\_process\_groups\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all process groups

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_process\_groups\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupsEntity If the method is called asynchronously, returns the request thread.

#### **get\_processors** (*id*, *\*\*kwargs*)

Gets all processors

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processors(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_descendant\_groups** (*bool*) – Whether or not to include processors from descendant process groups

**Returns** ProcessorsEntity If the method is called asynchronously, returns the request thread.

**get\_processors\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all processors

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processors\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_descendant\_groups** (*bool*) – Whether or not to include processors from descendant process groups

**Returns** ProcessorsEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_groups** (*id*, *\*\*kwargs*)

Gets all remote process groups

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_remote\_process\_groups(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** RemoteProcessGroupsEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_groups\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets all remote process groups

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_remote\_process\_groups\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** RemoteProcessGroupsEntity If the method is called asynchronously, returns the request thread.

**get\_replace\_process\_group\_request** (*id*, *\*\*kwargs*)

Returns the Replace Request with the given ID Returns the Replace Request with the given ID. Once a Replace Request has been created by performing a POST to /process-groups/{id}/replace-requests, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_replace\_process\_group\_request(id, callback=callback\_function)

**Parameters**



- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Replace Request (required)

**Returns** ProcessGroupReplaceRequestEntity If the method is called asynchronously, returns the request thread.

**get\_replace\_process\_group\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Returns the Replace Request with the given ID Returns the Replace Request with the given ID. Once a Replace Request has been created by performing a POST to /process-groups/{id}/replace-requests, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_replace\_process\_group\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Replace Request (required)

**Returns** ProcessGroupReplaceRequestEntity If the method is called asynchronously, returns the request thread.

**get\_variable\_registry** (*id*, *\*\*kwargs*)

Gets a process group's variable registry Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_variable\_registry(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_ancestor\_groups** (*bool*) – Whether or not to include ancestor groups

**Returns** VariableRegistryEntity If the method is called asynchronously, returns the request thread.

**get\_variable\_registry\_update\_request** (*group\_id*, *update\_id*, *\*\*kwargs*)

Gets a process group's variable registry Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_variable\_registry\_update\_request(group\_id, update\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **group\_id** (*str*) – The process group id. (required)
- **update\_id** (*str*) – The ID of the Variable Registry Update Request (required)

**Returns** VariableRegistryUpdateRequestEntity If the method is called asynchronously, returns the request thread.

```
get_variable_registry_update_request_with_http_info(group_id, update_id,
                                                    **kwargs)
```

Gets a process group's variable registry Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_variable\_registry\_update\_request\_with\_http\_info(group\_id, update\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **group\_id** (*str*) – The process group id. (required)
- **update\_id** (*str*) – The ID of the Variable Registry Update Request (required)

**Returns** VariableRegistryUpdateRequestEntity If the method is called asynchronously, returns the request thread.

```
get_variable_registry_with_http_info(id, **kwargs)
```

Gets a process group's variable registry Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_variable\_registry\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **include\_ancestor\_groups** (*bool*) – Whether or not to include ancestor groups

**Returns** VariableRegistryEntity If the method is called asynchronously, returns the request thread.

```
import_process_group(id, **kwargs)
```

Imports a specified process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_process\_group(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

```
import_process_group_with_http_info(id, **kwargs)
```

Imports a specified process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_process\_group\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The process group id. (required)

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**import\_template** (*id*, *\*\*kwargs*)

Imports a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_template(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

**import\_template\_with\_http\_info** (*id*, *\*\*kwargs*)

Imports a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_template\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

**initiate\_replace\_process\_group** (*id*, *body*, *\*\*kwargs*)

Initiate the Replace Request of a Process Group with the given ID This will initiate the action of replacing a process group with the given process group. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a ProcessGroupReplaceRequestEntity, and the process of replacing the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /process-groups/replace-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /process-groups/replace-requests/{requestId}. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.initiate\_replace\_process\_group(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupImportEntity*) – The process group replace request entity (required)

**Returns** ProcessGroupReplaceRequestEntity If the method is called asynchronously, returns the request thread.

**initiate\_replace\_process\_group\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Initiate the Replace Request of a Process Group with the given ID This will initiate the action of

replacing a process group with the given process group. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a `ProcessGroupReplaceRequestEntity`, and the process of replacing the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/process-groups/replace-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/process-groups/replace-requests/{requestId}`. Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.initiate_replace_process_group_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupImportEntity*) – The process group replace request entity (required)

**Returns** `ProcessGroupReplaceRequestEntity` If the method is called asynchronously, returns the request thread.

**instantiate\_template** (*id, body, \*\*kwargs*)

Instantiates a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.instantiate_template(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*InstantiateTemplateRequestEntity*) – The instantiate template request. (required)

**Returns** `FlowEntity` If the method is called asynchronously, returns the request thread.

**instantiate\_template\_with\_http\_info** (*id, body, \*\*kwargs*)

Instantiates a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.instantiate_template_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*InstantiateTemplateRequestEntity*) – The instantiate template request. (required)

**Returns** `FlowEntity` If the method is called asynchronously, returns the request thread.

**remove\_drop\_request** (*id, drop\_request\_id, \*\*kwargs*)

Cancels and/or removes a request to drop all flowfiles.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_drop\_request(id, drop\_request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_drop\_request\_with\_http\_info** (*id*, *drop\_request\_id*, *\*\*kwargs*)

Cancels and/or removes a request to drop all flowfiles.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_drop\_request\_with\_http\_info(id, drop\_request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **drop\_request\_id** (*str*) – The drop request id. (required)

**Returns** DropRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_process\_group** (*id*, *\*\*kwargs*)

Deletes a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_process\_group(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**remove\_process\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_process\_group\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**replace\_process\_group** (*id, body, \*\*kwargs*)

Replace Process Group contents with the given ID with the specified Process Group contents This endpoint is used for replication within a cluster, when replacing a flow with a new flow. It expects that the flow being replaced is not under version control and that the given snapshot will not modify any Processor that is currently running or any Controller Service that is enabled. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.replace\_process\_group(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupImportEntity*) – The process group replace request entity. (required)

**Returns** ProcessGroupImportEntity If the method is called asynchronously, returns the request thread.

**replace\_process\_group\_with\_http\_info** (*id, body, \*\*kwargs*)

Replace Process Group contents with the given ID with the specified Process Group contents This endpoint is used for replication within a cluster, when replacing a flow with a new flow. It expects that the flow being replaced is not under version control and that the given snapshot will not modify any Processor that is currently running or any Controller Service that is enabled. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.replace\_process\_group\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupImportEntity*) – The process group replace request entity. (required)

**Returns** ProcessGroupImportEntity If the method is called asynchronously, returns the request thread.

**submit\_update\_variable\_registry\_request** (*id, body, \*\*kwargs*)

Submits a request to update a process group's variable registry Note: This endpoint is subject to

change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_update\_variable\_registry\_request(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VariableRegistryEntity*) – The variable registry configuration details. (required)

**Returns** *VariableRegistryUpdateRequestEntity* If the method is called asynchronously, returns the request thread.

#### **submit\_update\_variable\_registry\_request\_with\_http\_info** (*id, body, \*\*kwargs*)

Submits a request to update a process group's variable registry Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_update\_variable\_registry\_request\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VariableRegistryEntity*) – The variable registry configuration details. (required)

**Returns** *VariableRegistryUpdateRequestEntity* If the method is called asynchronously, returns the request thread.

#### **update\_process\_group** (*id, body, \*\*kwargs*)

Updates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_process\_group(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupEntity*) – The process group configuration details. (required)

**Returns** *ProcessGroupEntity* If the method is called asynchronously, returns the request thread.

#### **update\_process\_group\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_process\_group\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The process group id. (required)
- **body** (*ProcessGroupEntity*) – The process group configuration details. (required)

**Returns** *ProcessGroupEntity* If the method is called asynchronously, returns the request thread.

**update\_variable\_registry** (*id*, *body*, *\*\*kwargs*)

Updates the contents of a Process Group’s variable Registry Note: This endpoint is subject to change as NiFi and it’s REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_variable\_registry(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VariableRegistryEntity*) – The variable registry configuration details. (required)

**Returns** *VariableRegistryEntity* If the method is called asynchronously, returns the request thread.

**update\_variable\_registry\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates the contents of a Process Group’s variable Registry Note: This endpoint is subject to change as NiFi and it’s REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_variable\_registry\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VariableRegistryEntity*) – The variable registry configuration details. (required)

**Returns** *VariableRegistryEntity* If the method is called asynchronously, returns the request thread.

**upload\_process\_group** (*id*, *body*, *body2*, *body3*, *body4*, *\*\*kwargs*)

Uploads a versioned flow definition and creates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.upload\_process\_group(id, body, body2, body3, body4, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*str*) – The process group name. (required)
- **body2** (*float*) – The process group X position. (required)
- **body3** (*float*) – The process group Y position. (required)
- **body4** (*str*) – The client id. (required)



- **body5** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**upload\_process\_group\_with\_http\_info** (*id, body, body2, body3, body4, \*\*kwargs*)

Uploads a versioned flow definition and creates a process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.upload\_process\_group\_with\_http\_info(id, body, body2, body3, body4, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*str*) – The process group name. (required)
- **body2** (*float*) – The process group X position. (required)
- **body3** (*float*) – The process group Y position. (required)
- **body4** (*str*) – The client id. (required)
- **body5** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessGroupEntity If the method is called asynchronously, returns the request thread.

**upload\_template** (*id, template, \*\*kwargs*)

Uploads a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.upload\_template(id, template, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **template** (*file*) – The binary content of the template file being uploaded. (required)
- **body** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

**upload\_template\_with\_http\_info** (*id, template, \*\*kwargs*)

Uploads a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.upload\_template\_with\_http\_info(id, template, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

- **template** (*file*) – The binary content of the template file being uploaded. (required)
- **body** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.processors\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.processors\_api.ProcessorsApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**analyze\_configuration** (*id, body, \*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ConfigurationAnalysisEntity*) – The processor configuration analysis request. (required)

**Returns** ConfigurationAnalysisEntity If the method is called asynchronously, returns the request thread.

**analyze\_configuration\_with\_http\_info** (*id, body, \*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ConfigurationAnalysisEntity*) – The processor configuration analysis request. (required)

**Returns** ConfigurationAnalysisEntity If the method is called asynchronously, returns the request thread.

**clear\_state** (*id*, *\*\*kwargs*)

Clears the state for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**clear\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Clears the state for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**delete\_processor** (*id*, *\*\*kwargs*)

Deletes a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_processor(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**delete\_processor\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_processor\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

#### **delete\_verification\_request** (*id, request\_id, \*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_verification\_request(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Processor (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

#### **delete\_verification\_request\_with\_http\_info** (*id, request\_id, \*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_verification\_request\_with\_http\_info(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Processor (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_processor** (*id*, *\*\*kwargs*)

Gets a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_diagnostics** (*id*, *\*\*kwargs*)

Gets diagnostics information about a processor Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_diagnostics(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_diagnostics\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets diagnostics information about a processor Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_diagnostics\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_run\_status\_details** (*\*\*kwargs*)

Submits a query to retrieve the run status details of all processors that are in the given list of Processor IDs

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_run\_status\_details(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*RunStatusDetailsRequestEntity*) – The request for the processors that should be included in the results

**Returns** ProcessorsRunStatusDetailsEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_run\_status\_details\_with\_http\_info** (*\*\*kwargs*)

Submits a query to retrieve the run status details of all processors that are in the given list of Processor IDs

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_run\_status\_details\_with\_http\_info(callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*RunStatusDetailsRequestEntity*) – The request for the processors that should be included in the results

**Returns** ProcessorsRunStatusDetailsEntity If the method is called asynchronously, returns the request thread.

**get\_processor\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_processor\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ProcessorEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor** (*id*, *property\_name*, *\*\*kwargs*)

Gets the descriptor for a processor property

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_property\_descriptor(id, property\_name, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **property\_name** (*str*) – The property name. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor\_with\_http\_info** (*id*, *property\_name*, *\*\*kwargs*)

Gets the descriptor for a processor property

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_property\_descriptor\_with\_http\_info(id, property\_name, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **property\_name** (*str*) – The property name. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_state** (*id*, *\*\*kwargs*)

Gets the state for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the state for a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request** (*id*, *request\_id*, *\*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID. Once an Verification Request has been created, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request(id, request\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Processor (required)

- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request\_with\_http\_info** (*id*, *request\_id*, *\*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID. Once an Verification Request has been created, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request\_with\_http\_info(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Processor (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**submit\_processor\_verification\_request** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Processor's configuration This will initiate the process of verifying a given Processor configuration. This may be a long-running task. As a result, this endpoint will immediately return a ProcessorConfigVerificationRequestEntity, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /processors/{processorId}/verification-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /processors/{processorId}/verification-requests/{requestId}. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_processor\_verification\_request(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*VerifyConfigRequestEntity*) – The processor configuration verification request. (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**submit\_processor\_verification\_request\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Processor's configuration This will initiate the process of verifying a given Processor configuration. This may be a long-running task. As a result, this endpoint will immediately return a ProcessorConfigVerificationRequestEntity, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /processors/{processorId}/verification-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /processors/{processorId}/verification-requests/{requestId}. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_processor\_verification\_request\_with\_http\_info(id, body, callback=callback\_function)



**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*VerifyConfigRequestEntity*) – The processor configuration verification request. (required)

**Returns** *VerifyConfigRequestEntity* If the method is called asynchronously, returns the request thread.

**terminate\_processor** (*id*, *\*\*kwargs*)

Terminates a processor, essentially “deleting” its threads and any active tasks

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.terminate\_processor(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**terminate\_processor\_with\_http\_info** (*id*, *\*\*kwargs*)

Terminates a processor, essentially “deleting” its threads and any active tasks

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.terminate\_processor\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**update\_processor** (*id*, *body*, *\*\*kwargs*)

Updates a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_processor(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ProcessorEntity*) – The processor configuration details. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**update\_processor\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_processor_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ProcessorEntity*) – The processor configuration details. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status** (*id, body, \*\*kwargs*)

Updates run status of a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_run_status(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ProcessorRunStatusEntity*) – The processor run status. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates run status of a processor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_run_status_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)
- **body** (*ProcessorRunStatusEntity*) – The processor run status. (required)

**Returns** *ProcessorEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.provenance\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.apis.provenance_api.ProvenanceApi (api_client=None)
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

```
delete_lineage (id, **kwargs)
```

Deletes a lineage query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_lineage(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the lineage query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

```
delete_lineage_with_http_info (id, **kwargs)
```

Deletes a lineage query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_lineage\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the lineage query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

```
delete_provenance (id, **kwargs)
```

Deletes a provenance query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_provenance(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the provenance query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

```
delete_provenance_with_http_info (id, **kwargs)
```

Deletes a provenance query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_provenance_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the provenance query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

**get\_lineage** (*id*, *\*\*kwargs*)

Gets a lineage query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_lineage(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the lineage query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

**get\_lineage\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a lineage query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_lineage_with_http_info(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the lineage query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

**get\_provenance** (*id*, *\*\*kwargs*)

Gets a provenance query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_provenance(id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the provenance query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.

- **summarize** (*bool*) – Whether or not incremental results are returned. If false, provenance events are only returned once the query completes. This property is true by default.
- **incremental\_results** (*bool*) – Whether or not to summarize provenance events returned. This property is false by default.

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

**get\_provenance\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a provenance query

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_provenance\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The id of the provenance query. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this query exists if clustered.
- **summarize** (*bool*) – Whether or not incremental results are returned. If false, provenance events are only returned once the query completes. This property is true by default.
- **incremental\_results** (*bool*) – Whether or not to summarize provenance events returned. This property is false by default.

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

**get\_search\_options** (*\*\*kwargs*)

Gets the searchable attributes for provenance events

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_search\_options(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ProvenanceOptionsEntity If the method is called asynchronously, returns the request thread.

**get\_search\_options\_with\_http\_info** (*\*\*kwargs*)

Gets the searchable attributes for provenance events

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_search\_options\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ProvenanceOptionsEntity If the method is called asynchronously, returns the request thread.

**submit\_lineage\_request** (*body*, *\*\*kwargs*)

Submits a lineage query Lineage queries may be long running so this endpoint submits a request. The response will include the current state of the query. If the request is not completed the URI in the response can be used at a later time to get the updated state of the query. Once the query has completed the lineage

request should be deleted by the client who originally submitted it. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_lineage\_request(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*LineageEntity*) – The lineage query details. (required)

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

#### **submit\_lineage\_request\_with\_http\_info** (*body*, *\*\*kwargs*)

Submits a lineage query Lineage queries may be long running so this endpoint submits a request. The response will include the current state of the query. If the request is not completed the URI in the response can be used at a later time to get the updated state of the query. Once the query has completed the lineage request should be deleted by the client who originally submitted it. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_lineage\_request\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*LineageEntity*) – The lineage query details. (required)

**Returns** LineageEntity If the method is called asynchronously, returns the request thread.

#### **submit\_provenance\_request** (*body*, *\*\*kwargs*)

Submits a provenance query Provenance queries may be long running so this endpoint submits a request. The response will include the current state of the query. If the request is not completed the URI in the response can be used at a later time to get the updated state of the query. Once the query has completed the provenance request should be deleted by the client who originally submitted it. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_provenance\_request(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ProvenanceEntity*) – The provenance query details. (required)

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

#### **submit\_provenance\_request\_with\_http\_info** (*body*, *\*\*kwargs*)

Submits a provenance query Provenance queries may be long running so this endpoint submits a request. The response will include the current state of the query. If the request is not completed the URI in the response can be used at a later time to get the updated state of the query. Once the query has completed the provenance request should be deleted by the client who originally submitted it. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_provenance\_request\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*ProvenanceEntity*) – The provenance query details. (required)

**Returns** ProvenanceEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.provenance\_events\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.provenance\_events\_api.ProvenanceEventsApi (api\_client=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_input\_content** (id, \*\*kwargs)

Gets the input content for a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_content(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_input\_content\_with\_http\_info** (id, \*\*kwargs)

Gets the input content for a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_input\_content\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_output\_content** (id, \*\*kwargs)

Gets the output content for a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_content(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_output\_content\_with\_http\_info** (*id*, **\*\*kwargs**)

Gets the output content for a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_output\_content\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where the content exists if clustered.

**Returns** StreamingOutput If the method is called asynchronously, returns the request thread.

**get\_provenance\_event** (*id*, **\*\*kwargs**)

Gets a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_provenance\_event(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this event exists if clustered.

**Returns** ProvenanceEventEntity If the method is called asynchronously, returns the request thread.

**get\_provenance\_event\_with\_http\_info** (*id*, **\*\*kwargs**)

Gets a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_provenance\_event\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The provenance event id. (required)
- **cluster\_node\_id** (*str*) – The id of the node where this event exists if clustered.

**Returns** ProvenanceEventEntity If the method is called asynchronously, returns the request thread.



**submit\_replay** (*body*, *\*\*kwargs*)

Replays content from a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_replay(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*SubmitReplayRequestEntity*) – The replay request. (required)

**Returns** *ProvenanceEventEntity* If the method is called asynchronously, returns the request thread.

**submit\_replay\_with\_http\_info** (*body*, *\*\*kwargs*)

Replays content from a provenance event

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_replay\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*SubmitReplayRequestEntity*) – The replay request. (required)

**Returns** *ProvenanceEventEntity* If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.remote\_process\_groups\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen>

**class** nipyapi.nifi.apis.remote\_process\_groups\_api.**RemoteProcessGroupsApi** (*api\_client=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_remote\_process\_group** (*id*, *\*\*kwargs*)

Gets a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_remote\_process\_group(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **id** (*str*) – The remote process group id. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**get\_remote\_process\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_remote\_process\_group\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**get\_state** (*id*, *\*\*kwargs*)

Gets the state for a RemoteProcessGroup

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the state for a RemoteProcessGroup

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The processor id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**remove\_remote\_process\_group** (*id*, *\*\*kwargs*)

Deletes a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_remote\_process\_group(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**remove\_remote\_process\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_remote\_process\_group\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group** (*id*, *body*, *\*\*kwargs*)

Updates a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **body** (RemoteProcessGroupEntity) – The remote process group. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_input\_port** (*id*, *port\_id*, *body*, *\*\*kwargs*)

Updates a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving

the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_input\_port(id, port\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemoteProcessGroupPortEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_input\_port\_run\_status** (*id, port\_id, body, \*\*kwargs*)

Updates run status of a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_input\_port\_run\_status(id, port\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_input\_port\_run\_status\_with\_http\_info** (*id, port\_id, body, \*\*kwargs*)

Updates run status of a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_input\_port\_run\_status\_with\_http\_info(id, port\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_input\_port\_with\_http\_info** (*id, port\_id, body, \*\*kwargs*)

Updates a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an

asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_remote_process_group_input_port_with_http_info(id, port_id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemoteProcessGroupPortEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_output\_port** (*id, port\_id, body, \*\*kwargs*)

Updates a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_remote_process_group_output_port(id, port_id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemoteProcessGroupPortEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_output\_port\_run\_status** (*id, port\_id, body, \*\*kwargs*)

Updates run status of a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_remote_process_group_output_port_run_status(id, port_id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_output\_port\_run\_status\_with\_http\_info** (*id, port\_id, body, \*\*kwargs*)

Updates run status of a remote port Note: This endpoint is subject to change as NiFi and it's

REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_output\_port\_run\_status\_with\_http\_info(id, port\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_output\_port\_with\_http\_info** (*id*, *port\_id*, *body*, *\*\*kwargs*)

Updates a remote port Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_output\_port\_with\_http\_info(id, port\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **port\_id** (*str*) – The remote process group port id. (required)
- **body** (*RemoteProcessGroupPortEntity*) – The remote process group port. (required)

**Returns** RemoteProcessGroupPortEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_run\_status** (*id*, *body*, *\*\*kwargs*)

Updates run status of a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_remote\_process\_group\_run\_status(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group run status. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_run\_status\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates run status of a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.update_remote_process_group_run_status_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process group run status. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_run\_statuses** (*id, body, \*\*kwargs*)

Updates run status of all remote process groups in a process group (recursively)

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.update_remote_process_group_run_statuses(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process groups run status. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_run\_statuses\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates run status of all remote process groups in a process group (recursively)

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.update_remote_process_group_run_statuses_with_http_info(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*RemotePortRunStatusEntity*) – The remote process groups run status. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**update\_remote\_process\_group\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates a remote process group

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response.

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.update_remote_process_group_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The remote process group id. (required)
- **body** (*RemoteProcessGroupEntity*) – The remote process group. (required)

**Returns** RemoteProcessGroupEntity If the method is called asynchronously, returns the request thread.

**nipyapi.nifi.apis.reporting\_tasks\_api module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.reporting\_tasks\_api.**ReportingTasksApi** (*api\_client=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**analyze\_configuration** (*id, body, \*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*ConfigurationAnalysisEntity*) – The configuration analysis request. (required)

**Returns** ConfigurationAnalysisEntity If the method is called asynchronously, returns the request thread.

**analyze\_configuration\_with\_http\_info** (*id, body, \*\*kwargs*)

Performs analysis of the component's configuration, providing information about which attributes are referenced.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.analyze\_configuration\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)



- **body** (*ConfigurationAnalysisEntity*) – The configuration analysis request. (required)

**Returns** *ConfigurationAnalysisEntity* If the method is called asynchronously, returns the request thread.

**clear\_state** (*id*, *\*\*kwargs*)

Clears the state for a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** *ComponentStateEntity* If the method is called asynchronously, returns the request thread.

**clear\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Clears the state for a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.clear\_state\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** *ComponentStateEntity* If the method is called asynchronously, returns the request thread.

**delete\_validation\_request** (*id*, *request\_id*, *\*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_validation\_request(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Reporting Task (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** *VerifyConfigRequestEntity* If the method is called asynchronously, returns the request thread.

**delete\_validation\_request\_with\_http\_info** (*id*, *request\_id*, *\*\*kwargs*)

Deletes the Verification Request with the given ID Deletes the Verification Request with the given ID. After a request is created, it is expected that the client will properly clean up the request by

DELETE'ing it, once the Verification process has completed. If the request is deleted before the request completes, then the Verification request will finish the step that it is currently performing and then will cancel any subsequent steps. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_validation_request_with_http_info(id, request_id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Reporting Task (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor** (*id, property\_name, \*\*kwargs*)

Gets a reporting task property descriptor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_property_descriptor(id, property_name, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **property\_name** (*str*) – The property name. (required)
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_property\_descriptor\_with\_http\_info** (*id, property\_name, \*\*kwargs*)

Gets a reporting task property descriptor

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.get_property_descriptor_with_http_info(id, property_name, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **property\_name** (*str*) – The property name. (required)
- **sensitive** (*bool*) – Property Descriptor requested sensitive status

**Returns** PropertyDescriptorEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_task** (*id, \*\*kwargs*)

Gets a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def
```

```
callback_function(response): >>> pprint(response) >>> >>> thread = api.get_reporting_task(id, call-
back=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** ReportingTaskEntity If the method is called asynchronously, returns the request thread.

**get\_reporting\_task\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_reporting\_task\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** ReportingTaskEntity If the method is called asynchronously, returns the request thread.

**get\_state** (*id*, *\*\*kwargs*)

Gets the state for a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_state\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the state for a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_state\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)

**Returns** ComponentStateEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request** (*id*, *request\_id*, *\*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID.

Once an Verification Request has been created, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Reporting Task (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**get\_verification\_request\_with\_http\_info** (*id, request\_id, \*\*kwargs*)

Returns the Verification Request with the given ID Returns the Verification Request with the given ID. Once an Verification Request has been created, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_verification\_request\_with\_http\_info(id, request\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Reporting Task (required)
- **request\_id** (*str*) – The ID of the Verification Request (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**remove\_reporting\_task** (*id, \*\*kwargs*)

Deletes a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_reporting\_task(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ReportingTaskEntity If the method is called asynchronously, returns the request thread.

**remove\_reporting\_task\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_reporting\_task\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** ReportingTaskEntity If the method is called asynchronously, returns the request thread.

**submit\_config\_verification\_request** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Reporting Task's configuration This will initiate the process of verifying a given Reporting Task configuration. This may be a long-running task. As a result, this endpoint will immediately return a ReportingTaskConfigVerificationRequestEntity, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /reporting-tasks/{serviceId}/verification-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /reporting-tasks/{serviceId}/verification-requests/{requestId}. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_config\_verification\_request(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*VerifyConfigRequestEntity*) – The reporting task configuration verification request. (required)

**Returns** VerifyConfigRequestEntity If the method is called asynchronously, returns the request thread.

**submit\_config\_verification\_request\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Performs verification of the Reporting Task's configuration This will initiate the process of verifying a given Reporting Task configuration. This may be a long-running task. As a result, this endpoint will immediately return a ReportingTaskConfigVerificationRequestEntity, and the process of performing the verification will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /reporting-tasks/{serviceId}/verification-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /reporting-tasks/{serviceId}/verification-requests/{requestId}. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.submit\_config\_verification\_request\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*VerifyConfigRequestEntity*) – The reporting task configuration verification request. (required)

**Returns** *VerifyConfigRequestEntity* If the method is called asynchronously, returns the request thread.

**update\_reporting\_task** (*id*, *body*, *\*\*kwargs*)

Updates a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_reporting\_task(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*ReportingTaskEntity*) – The reporting task configuration details. (required)

**Returns** *ReportingTaskEntity* If the method is called asynchronously, returns the request thread.

**update\_reporting\_task\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_reporting\_task\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*ReportingTaskEntity*) – The reporting task configuration details. (required)

**Returns** *ReportingTaskEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status** (*id*, *body*, *\*\*kwargs*)

Updates run status of a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*ReportingTaskRunStatusEntity*) – The reporting task run status. (required)

**Returns** *ReportingTaskEntity* If the method is called asynchronously, returns the request thread.

**update\_run\_status\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates run status of a reporting task

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_run\_status\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The reporting task id. (required)
- **body** (*ReportingTaskRunStatusEntity*) – The reporting task run status. (required)

**Returns** ReportingTaskEntity If the method is called asynchronously, returns the request thread.

### nipyapi.nifi.apis.resources\_api module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.resources\_api.**ResourcesApi** (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_resources** (*\*\*kwargs*)

Gets the available resources that support access/authorization policies

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_resources(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ResourcesEntity If the method is called asynchronously, returns the request thread.

**get\_resources\_with\_http\_info** (*\*\*kwargs*)

Gets the available resources that support access/authorization policies

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_resources\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ResourcesEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.site\_to\_site\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.site\_to\_site\_api.SiteToSiteApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_peers** (\*\*kwargs)

Returns the available Peers and its status of this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_peers(callback=callback\_function)

**Parameters function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** PeersEntity If the method is called asynchronously, returns the request thread.

**get\_peers\_with\_http\_info** (\*\*kwargs)

Returns the available Peers and its status of this NiFi

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_peers\_with\_http\_info(callback=callback\_function)

**Parameters function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** PeersEntity If the method is called asynchronously, returns the request thread.

**get\_site\_to\_site\_details** (\*\*kwargs)

Returns the details about this NiFi necessary to communicate via site to site

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_site\_to\_site\_details(callback=callback\_function)

**Parameters function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** ControllerEntity If the method is called asynchronously, returns the request thread.

**get\_site\_to\_site\_details\_with\_http\_info** (\*\*kwargs)

Returns the details about this NiFi necessary to communicate via site to site

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_site\_to\_site\_details\_with\_http\_info(callback=callback\_function)



**Parameters** `function` (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `ControllerEntity` If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.snippets\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.apis.snippets_api.SnippetsApi` (*api\_client=None*)

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_snippet** (*body, \*\*kwargs*)

Creates a snippet. The snippet will be automatically discarded if not used in a subsequent request after 1 minute.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_snippet(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (`SnippetEntity`) – The snippet configuration details. (required)

**Returns** `SnippetEntity` If the method is called asynchronously, returns the request thread.

**create\_snippet\_with\_http\_info** (*body, \*\*kwargs*)

Creates a snippet. The snippet will be automatically discarded if not used in a subsequent request after 1 minute.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_snippet\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (`SnippetEntity`) – The snippet configuration details. (required)

**Returns** `SnippetEntity` If the method is called asynchronously, returns the request thread.

**delete\_snippet** (*id, \*\*kwargs*)

Deletes the components in a snippet and discards the snippet

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def

```
callback_function(response): >>> pprint(response) >>> >>> thread = api.delete_snippet(id, call-  
back=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The snippet id. (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** SnippetEntity If the method is called asynchronously, returns the request thread.

**delete\_snippet\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes the components in a snippet and discards the snippet

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_snippet\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The snippet id. (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** SnippetEntity If the method is called asynchronously, returns the request thread.

**update\_snippet** (*id*, *body*, *\*\*kwargs*)

Move's the components in this Snippet into a new Process Group and discards the snippet

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_snippet(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The snippet id. (required)
- **body** (*SnippetEntity*) – The snippet configuration details. (required)

**Returns** SnippetEntity If the method is called asynchronously, returns the request thread.

**update\_snippet\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Move's the components in this Snippet into a new Process Group and discards the snippet

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_snippet\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The snippet id. (required)
- **body** (*SnippetEntity*) – The snippet configuration details. (required)

**Returns** SnippetEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.system\_diagnostics\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.system\_diagnostics\_api.**SystemDiagnosticsApi** (*api\_client=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_system\_diagnostics** (*\*\*kwargs*)

Gets the diagnostics for the system NiFi is running on

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_system\_diagnostics(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** SystemDiagnosticsEntity If the method is called asynchronously, returns the request thread.

**get\_system\_diagnostics\_with\_http\_info** (*\*\*kwargs*)

Gets the diagnostics for the system NiFi is running on

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_system\_diagnostics\_with\_http\_info(callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **nodewise** (*bool*) – Whether or not to include the breakdown per node. Optional, defaults to false
- **cluster\_node\_id** (*str*) – The id of the node where to get the status.

**Returns** SystemDiagnosticsEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.templates\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.templates\_api.TemplatesApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**export\_template** (id, \*\*kwargs)

Exports a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_template(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The template id. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**export\_template\_with\_http\_info** (id, \*\*kwargs)

Exports a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_template\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The template id. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**remove\_template** (id, \*\*kwargs)

Deletes a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_template(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The template id. (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

**remove\_template\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a template

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_template\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The template id. (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** TemplateEntity If the method is called asynchronously, returns the request thread.

### nipyapi.nifi.apis.tenants\_api module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.tenants\_api.TenantsApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_user** (*body*, *\*\*kwargs*)

Creates a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*UserEntity*) – The user configuration details. (required)

**Returns** UserEntity If the method is called asynchronously, returns the request thread.

**create\_user\_group** (*body*, *\*\*kwargs*)

Creates a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user\_group(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **body** (`UserGroupEntity`) – The user group configuration details. (required)

**Returns** `UserGroupEntity` If the method is called asynchronously, returns the request thread.

**create\_user\_group\_with\_http\_info** (*body*, *\*\*kwargs*)

Creates a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> thread = api.create\_user\_group\_with\_http\_info(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (`UserGroupEntity`) – The user group configuration details. (required)

**Returns** `UserGroupEntity` If the method is called asynchronously, returns the request thread.

**create\_user\_with\_http\_info** (*body*, *\*\*kwargs*)

Creates a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> thread = api.create\_user\_with\_http\_info(body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (`UserEntity`) – The user configuration details. (required)

**Returns** `UserEntity` If the method is called asynchronously, returns the request thread.

**get\_user** (*id*, *\*\*kwargs*)

Gets a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> thread = api.get\_user(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)

**Returns** `UserEntity` If the method is called asynchronously, returns the request thread.

**get\_user\_group** (*id*, *\*\*kwargs*)

Gets a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> thread = api.get\_user\_group(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)

**Returns** `UserGroupEntity` If the method is called asynchronously, returns the request thread.

**get\_user\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a

*callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_group\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)

**Returns** UserGroupEntity If the method is called asynchronously, returns the request thread.

#### get\_user\_groups (\*\*kwargs)

Gets all user groups Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_groups(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** UserGroupsEntity If the method is called asynchronously, returns the request thread.

#### get\_user\_groups\_with\_http\_info (\*\*kwargs)

Gets all user groups Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_groups\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** UserGroupsEntity If the method is called asynchronously, returns the request thread.

#### get\_user\_with\_http\_info (id, \*\*kwargs)

Gets a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)

**Returns** UserEntity If the method is called asynchronously, returns the request thread.

#### get\_users (\*\*kwargs)

Gets all users Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_users(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** UsersEntity If the method is called asynchronously, returns the request thread.

#### get\_users\_with\_http\_info (\*\*kwargs)

Gets all users Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a

*callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_users\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** UsersEntity If the method is called asynchronously, returns the request thread.

**remove\_user** (*id*, **\*\*kwargs**)

Deletes a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** UserEntity If the method is called asynchronously, returns the request thread.

**remove\_user\_group** (*id*, **\*\*kwargs**)

Deletes a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user\_group(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** UserGroupEntity If the method is called asynchronously, returns the request thread.

**remove\_user\_group\_with\_http\_info** (*id*, **\*\*kwargs**)

Deletes a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user\_group\_with\_http\_info(id, callback=callback\_function)

**Parameters**



- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** UserGroupEntity If the method is called asynchronously, returns the request thread.

**remove\_user\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **version** (*str*) – The revision is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** UserEntity If the method is called asynchronously, returns the request thread.

**search\_tenants** (*q*, *\*\*kwargs*)

Searches for a tenant with the specified identity Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.search\_tenants(q, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) – Identity to search for. (required)

**Returns** TenantsEntity If the method is called asynchronously, returns the request thread.

**search\_tenants\_with\_http\_info** (*q*, *\*\*kwargs*)

Searches for a tenant with the specified identity Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.search\_tenants\_with\_http\_info(q, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **q** (*str*) – Identity to search for. (required)

**Returns** `TenantsEntity` If the method is called asynchronously, returns the request thread.

**update\_user** (*id*, *body*, *\*\*kwargs*)

Updates a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **body** (`UserEntity`) – The user configuration details. (required)

**Returns** `UserEntity` If the method is called asynchronously, returns the request thread.

**update\_user\_group** (*id*, *body*, *\*\*kwargs*)

Updates a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user_group(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **body** (`UserGroupEntity`) – The user group configuration details. (required)

**Returns** `UserGroupEntity` If the method is called asynchronously, returns the request thread.

**update\_user\_group\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a user group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user_group_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **body** (`UserGroupEntity`) – The user group configuration details. (required)

**Returns** `UserGroupEntity` If the method is called asynchronously, returns the request thread.

**update\_user\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Updates a user Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **body** (`UserEntity`) – The user configuration details. (required)

**Returns** UserEntity If the method is called asynchronously, returns the request thread.

## nipyapi.nifi.apis.versions\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.apis.versions\_api.VersionsApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_version\_control\_request** (body, \*\*kwargs)

Create a version control request Creates a request so that a Process Group can be placed under Version Control or have its Version Control configuration changed. Creating this request will prevent any other threads from simultaneously saving local changes to Version Control. It will not, however, actually save the local flow to the Flow Registry. A POST to /versions/process-groups/{id} should be used to initiate saving of the local flow to the Flow Registry. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_version\_control\_request(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*CreateActiveRequestEntity*) – The versioned flow details. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_version\_control\_request\_with\_http\_info** (body, \*\*kwargs)

Create a version control request Creates a request so that a Process Group can be placed under Version Control or have its Version Control configuration changed. Creating this request will prevent any other threads from simultaneously saving local changes to Version Control. It will not, however, actually save the local flow to the Flow Registry. A POST to /versions/process-groups/{id} should be used to initiate saving of the local flow to the Flow Registry. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_version\_control\_request\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*CreateActiveRequestEntity*) – The versioned flow details. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**delete\_revert\_request** (id, \*\*kwargs)

Deletes the Revert Request with the given ID Deletes the Revert Request with the given ID. After a request is created via a POST to /versions/revert-requests/process-groups/{id}, it is expected that the client will

properly clean up the request by DELETE'ing it, once the Revert process has completed. If the request is deleted before the request completes, then the Revert request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_revert\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Revert Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### `delete_revert_request_with_http_info` (*id*, *\*\*kwargs*)

Deletes the Revert Request with the given ID Deletes the Revert Request with the given ID. After a request is created via a POST to /versions/revert-requests/process-groups/{id}, it is expected that the client will properly clean up the request by DELETE'ing it, once the Revert process has completed. If the request is deleted before the request completes, then the Revert request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_revert\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Revert Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### `delete_update_request` (*id*, *\*\*kwargs*)

Deletes the Update Request with the given ID Deletes the Update Request with the given ID. After a request is created via a POST to /versions/update-requests/process-groups/{id}, it is expected that the client will properly clean up the request by DELETE'ing it, once the Update process has completed. If the request is deleted before the request completes, then the Update request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_update\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_update\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes the Update Request with the given ID Deletes the Update Request with the given ID. After a request is created via a POST to /versions/update-requests/process-groups/{id}, it is expected that the client will properly clean up the request by DELETE'ing it, once the Update process has completed. If the request is deleted before the request completes, then the Update request will finish the step that it is currently performing and then will cancel any subsequent steps. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_update\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**delete\_version\_control\_request** (*id*, *\*\*kwargs*)

Deletes the version control request with the given ID Deletes the Version Control Request with the given ID. This will allow other threads to save flows to the Flow Registry. See also the documentation for POSTing to /versions/active-requests for information regarding why this is done. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_version\_control\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The request ID. (required)
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** None If the method is called asynchronously, returns the request thread.

**delete\_version\_control\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Deletes the version control request with the given ID Deletes the Version Control Request with the given ID. This will allow other threads to save flows to the Flow Registry. See also the documentation for POSTing to /versions/active-requests for information regarding why this is done. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_version\_control\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The request ID. (required)

- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** None If the method is called asynchronously, returns the request thread.

**export\_flow\_version** (*id*, *\*\*kwargs*)

Gets the latest version of a Process Group for download

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_flow\_version(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**export\_flow\_version\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the latest version of a Process Group for download

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_flow\_version\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** str If the method is called asynchronously, returns the request thread.

**get\_revert\_request** (*id*, *\*\*kwargs*)

Returns the Revert Request with the given ID Returns the Revert Request with the given ID. Once a Revert Request has been created by performing a POST to /versions/revert-requests/process-groups/{id}, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_revert\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Revert Request (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**get\_revert\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Returns the Revert Request with the given ID Returns the Revert Request with the given ID. Once a Revert Request has been created by performing a POST to /versions/revert-requests/process-groups/{id}, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to

be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>>  
>>> thread = api.get\_revert\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Revert Request (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### **get\_update\_request** (*id*, *\*\*kwargs*)

Returns the Update Request with the given ID Returns the Update Request with the given ID. Once an Update Request has been created by performing a POST to /versions/update-requests/process-groups/{id}, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>>  
>>> thread = api.get\_update\_request(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### **get\_update\_request\_with\_http\_info** (*id*, *\*\*kwargs*)

Returns the Update Request with the given ID Returns the Update Request with the given ID. Once an Update Request has been created by performing a POST to /versions/update-requests/process-groups/{id}, that request can subsequently be retrieved via this endpoint, and the request that is fetched will contain the updated state, such as percent complete, the current state of the request, and any failures. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>>  
>>> thread = api.get\_update\_request\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The ID of the Update Request (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

#### **get\_version\_information** (*id*, *\*\*kwargs*)

Gets the Version Control information for a process group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_version\_information(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**get\_version\_information\_with\_http\_info** (*id*, *\*\*kwargs*)

Gets the Version Control information for a process group Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_version\_information\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**initiate\_revert\_flow\_version** (*id*, *body*, *\*\*kwargs*)

Initiate the Revert Request of a Process Group with the given ID For a Process Group that is already under Version Control, this will initiate the action of reverting any local changes that have been made to the Process Group since it was last synchronized with the Flow Registry. This will result in the flow matching the Versioned Flow that exists in the Flow Registry. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a VersionedFlowUpdateRequestEntity, and the process of updating the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /versions/revert-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /versions/revert-requests/{requestId}. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.initiate\_revert\_flow\_version(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionControlInformationEntity*) – The controller service configuration details. (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**initiate\_revert\_flow\_version\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Initiate the Revert Request of a Process Group with the given ID For a Process Group that is already under Version Control, this will initiate the action of reverting any local changes that have been made to the Process Group since it was last synchronized with the Flow Registry. This will result in the flow matching the Versioned Flow that exists in the Flow Registry. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a VersionedFlowUpdateRequestEntity, and the process of updating the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to /versions/revert-requests/{requestId}. Once the request is completed, the client is expected to issue a DELETE request to /versions/revert-requests/{requestId}. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.initiate\_revert\_flow\_version\_with\_http\_info(id, body, callback=callback\_function)



**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionControlInformationEntity*) – The controller service configuration details. (required)

**Returns** *VersionedFlowUpdateRequestEntity* If the method is called asynchronously, returns the request thread.

**initiate\_version\_control\_update** (*id, body, \*\*kwargs*)

Initiate the Update Request of a Process Group with the given ID For a Process Group that is already under Version Control, this will initiate the action of changing from a specific version of the flow in the Flow Registry to a different version of the flow. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a *VersionedFlowUpdateRequestEntity*, and the process of updating the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/versions/update-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/versions/update-requests/{requestId}`. Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> thread = api.initiate_version_control_update(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionControlInformationEntity*) – The controller service configuration details. (required)

**Returns** *VersionedFlowUpdateRequestEntity* If the method is called asynchronously, returns the request thread.

**initiate\_version\_control\_update\_with\_http\_info** (*id, body, \*\*kwargs*)

Initiate the Update Request of a Process Group with the given ID For a Process Group that is already under Version Control, this will initiate the action of changing from a specific version of the flow in the Flow Registry to a different version of the flow. This can be a lengthy process, as it will stop any Processors and disable any Controller Services necessary to perform the action and then restart them. As a result, the endpoint will immediately return a *VersionedFlowUpdateRequestEntity*, and the process of updating the flow will occur asynchronously in the background. The client may then periodically poll the status of the request by issuing a GET request to `/versions/update-requests/{requestId}`. Once the request is completed, the client is expected to issue a DELETE request to `/versions/update-requests/{requestId}`. Note: This endpoint is subject to change as NiFi and its REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> thread = api.initiate_version_control_update_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionControlInformationEntity*) – The controller service configuration details. (required)

**Returns** VersionedFlowUpdateRequestEntity If the method is called asynchronously, returns the request thread.

**save\_to\_flow\_registry** (*id*, *body*, *\*\*kwargs*)

Save the Process Group with the given ID Begins version controlling the Process Group with the given ID or commits changes to the Versioned Flow, depending on if the provided VersionControlInformation includes a flowId. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.save\_to\_flow\_registry(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*StartVersionControlRequestEntity*) – The versioned flow details. (required)

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**save\_to\_flow\_registry\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Save the Process Group with the given ID Begins version controlling the Process Group with the given ID or commits changes to the Versioned Flow, depending on if the provided VersionControlInformation includes a flowId. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.save\_to\_flow\_registry\_with\_http\_info(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*StartVersionControlRequestEntity*) – The versioned flow details. (required)

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**stop\_version\_control** (*id*, *\*\*kwargs*)

Stops version controlling the Process Group with the given ID Stops version controlling the Process Group with the given ID. The Process Group will no longer track to any Versioned Flow. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.stop\_version\_control(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, a new one will be generated. This value (whether specified or generated) is included in the response.

- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**stop\_version\_control\_with\_http\_info** (*id*, *\*\*kwargs*)

Stops version controlling the Process Group with the given ID Stops version controlling the Process Group with the given ID. The Process Group will no longer track to any Versioned Flow. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.stop\_version\_control\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the flow.
- **client\_id** (*str*) – If the client id is not specified, a new one will be generated. This value (whether specified or generated) is included in the response.
- **disconnected\_node\_acknowledged** (*bool*) – Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**update\_flow\_version** (*id*, *body*, *\*\*kwargs*)

Update the version of a Process Group with the given ID For a Process Group that is already under Version Control, this will update the version of the flow to a different version. This endpoint expects that the given snapshot will not modify any Processor that is currently running or any Controller Service that is enabled. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_flow\_version(id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionedFlowSnapshotEntity*) – The controller service configuration details. (required)

**Returns** VersionControlInformationEntity If the method is called asynchronously, returns the request thread.

**update\_flow\_version\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Update the version of a Process Group with the given ID For a Process Group that is already under Version Control, this will update the version of the flow to a different version. This endpoint expects that the given snapshot will not modify any Processor that is currently running or any Controller Service that is enabled. Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_flow\_version\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The process group id. (required)
- **body** (*VersionedFlowSnapshotEntity*) – The controller service configuration details. (required)

**Returns** *VersionControlInformationEntity* If the method is called asynchronously, returns the request thread.

**update\_version\_control\_request** (*id, body, \*\*kwargs*)

Updates the request with the given ID Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_version\_control\_request(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The request ID. (required)
- **body** (*VersionControlComponentMappingEntity*) – The version control component mapping. (required)

**Returns** *VersionControlInformationEntity* If the method is called asynchronously, returns the request thread.

**update\_version\_control\_request\_with\_http\_info** (*id, body, \*\*kwargs*)

Updates the request with the given ID Note: This endpoint is subject to change as NiFi and it's REST API evolve. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_version\_control\_request\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The request ID. (required)
- **body** (*VersionControlComponentMappingEntity*) – The version control component mapping. (required)

**Returns** *VersionControlInformationEntity* If the method is called asynchronously, returns the request thread.

**nipyapi.nifi.models package****Submodules****nipyapi.nifi.models.about\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.about_dto.AboutDTO (title=None, version=None, uri=None,
                                             content_viewer_url=None, time-
                                             zone=None, build_tag=None,
                                             build_revision=None, build_branch=None,
                                             build_timestamp=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AboutDTO - a model defined in Swagger

```
attribute_map = {'build_branch': 'buildBranch', 'build_revision': 'buildRevision', 'build_tag': 'buildTag', 'build_timestamp': 'buildTimestamp', 'content_viewer_url': 'contentViewerUrl', 'title': 'title', 'uri': 'uri', 'version': 'version', 'timezone': 'timezone'}
```

**build\_branch**

Gets the build\_branch of this AboutDTO. Build branch

**Returns** The build\_branch of this AboutDTO.

**Return type** str

**build\_revision**

Gets the build\_revision of this AboutDTO. Build revision or commit hash

**Returns** The build\_revision of this AboutDTO.

**Return type** str

**build\_tag**

Gets the build\_tag of this AboutDTO. Build tag

**Returns** The build\_tag of this AboutDTO.

**Return type** str

**build\_timestamp**

Gets the build\_timestamp of this AboutDTO. Build timestamp

**Returns** The build\_timestamp of this AboutDTO.

**Return type** str

**content\_viewer\_url**

Gets the content\_viewer\_url of this AboutDTO. The URL for the content viewer if configured.

**Returns** The content\_viewer\_url of this AboutDTO.

**Return type** str

```
swagger_types = {'build_branch': 'str', 'build_revision': 'str', 'build_tag': 'str', 'build_timestamp': 'str', 'content_viewer_url': 'str', 'title': 'str', 'uri': 'str', 'version': 'str', 'timezone': 'str'}
```

**timezone**

Gets the timezone of this AboutDTO. The timezone of the NiFi instance.

**Returns** The timezone of this AboutDTO.

**Return type** str

**title**

Gets the title of this AboutDTO. The title to be used on the page and in the about dialog.

**Returns** The title of this AboutDTO.

**Return type** str

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

**uri**  
Gets the uri of this AboutDTO. The URI for the NiFi.

**Returns** The uri of this AboutDTO.

**Return type** str

**version**  
Gets the version of this AboutDTO. The version of this NiFi.

**Returns** The version of this AboutDTO.

**Return type** str

### **nipyapi.nifi.models.about\_entity module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.about\_entity.**AboutEntity**(*about=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AboutEntity - a model defined in Swagger

**about**  
Gets the about of this AboutEntity.

**Returns** The about of this AboutEntity.

**Return type** *AboutDTO*

**attribute\_map** = {'about': 'about'}

**swagger\_types** = {'about': 'AboutDTO'}

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

### **nipyapi.nifi.models.access\_configuration\_dto module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.models.access_configuration_dto.AccessConfigurationDTO` (*supports\_login=None*)  
Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessConfigurationDTO - a model defined in Swagger

**attribute\_map** = {'supports\_login': 'supportsLogin'}

**supports\_login**

Gets the supports\_login of this AccessConfigurationDTO. Indicates whether or not this NiFi supports user login.

**Returns** The supports\_login of this AccessConfigurationDTO.

**Return type** `bool`

**swagger\_types** = {'supports\_login': 'bool'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

### nipyapi.nifi.models.access\_configuration\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.models.access_configuration_entity.AccessConfigurationEntity` (*config=None*)  
Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessConfigurationEntity - a model defined in Swagger

**attribute\_map** = {'config': 'config'}

**config**

Gets the config of this AccessConfigurationEntity.

**Returns** The config of this AccessConfigurationEntity.

**Return type** `AccessConfigurationDTO`

**swagger\_types** = {'config': 'AccessConfigurationDTO'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.access\_policy\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_policy_dto.AccessPolicyDTO (id=None,          ver-
                                                             sioned_component_id=None,
                                                             par-
                                                             ent_group_id=None,
                                                             position=None,
                                                             resource=None, ac-
                                                             tion=None, compo-
                                                             nent_reference=None,
                                                             configurable=None,
                                                             users=None,
                                                             user_groups=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicyDTO - a model defined in Swagger

#### **action**

Gets the action of this AccessPolicyDTO. The action associated with this access policy.

**Returns** The action of this AccessPolicyDTO.

**Return type** str

```
attribute_map = {'action':  'action', 'component_reference':  'componentReference', 'c
```

#### **component\_reference**

Gets the component\_reference of this AccessPolicyDTO. Component this policy references if applicable.

**Returns** The component\_reference of this AccessPolicyDTO.

**Return type** *ComponentReferenceEntity*

#### **configurable**

Gets the configurable of this AccessPolicyDTO. Whether this policy is configurable.

**Returns** The configurable of this AccessPolicyDTO.

**Return type** bool

#### **id**

Gets the id of this AccessPolicyDTO. The id of the component.

**Returns** The id of this AccessPolicyDTO.

**Return type** str

#### **parent\_group\_id**

Gets the parent\_group\_id of this AccessPolicyDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this AccessPolicyDTO.



**Return type** str

#### **position**

Gets the position of this AccessPolicyDTO. The position of this component in the UI if applicable.

**Returns** The position of this AccessPolicyDTO.

**Return type** *PositionDTO*

#### **resource**

Gets the resource of this AccessPolicyDTO. The resource for this access policy.

**Returns** The resource of this AccessPolicyDTO.

**Return type** str

**swagger\_types** = {'action': 'str', 'component\_reference': 'ComponentReferenceEntity',

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

#### **user\_groups**

Gets the user\_groups of this AccessPolicyDTO. The set of user group IDs associated with this access policy.

**Returns** The user\_groups of this AccessPolicyDTO.

**Return type** list[*TenantEntity*]

#### **users**

Gets the users of this AccessPolicyDTO. The set of user IDs associated with this access policy.

**Returns** The users of this AccessPolicyDTO.

**Return type** list[*TenantEntity*]

#### **versioned\_component\_id**

Gets the versioned\_component\_id of this AccessPolicyDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this AccessPolicyDTO.

**Return type** str

## **nipyapi.nifi.models.access\_policy\_entity module**

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity (revision=None,
                                                                    id=None,
                                                                    uri=None,
                                                                    position=None,
                                                                    permissions=None,
                                                                    bulletins=None,
                                                                    disconnected_node_acknowledged=None,
                                                                    generated=None,
                                                                    component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicyEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'component': 'component', 'disconnected\_node\_acknowledged': 'disconnected\_node\_acknowledged'}

#### **bulletins**

Gets the bulletins of this AccessPolicyEntity. The bulletins for this component.

**Returns** The bulletins of this AccessPolicyEntity.

**Return type** list[[BulletinEntity](#)]

#### **component**

Gets the component of this AccessPolicyEntity.

**Returns** The component of this AccessPolicyEntity.

**Return type** [AccessPolicyDTO](#)

#### **disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this AccessPolicyEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this AccessPolicyEntity.

**Return type** bool

#### **generated**

Gets the generated of this AccessPolicyEntity. When this content was generated.

**Returns** The generated of this AccessPolicyEntity.

**Return type** str

#### **id**

Gets the id of this AccessPolicyEntity. The id of the component.

**Returns** The id of this AccessPolicyEntity.

**Return type** str

#### **permissions**

Gets the permissions of this AccessPolicyEntity. The permissions for this component.

**Returns** The permissions of this AccessPolicyEntity.

**Return type** [PermissionsDTO](#)

**position**

Gets the position of this AccessPolicyEntity. The position of this component in the UI if applicable.

**Returns** The position of this AccessPolicyEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this AccessPolicyEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this AccessPolicyEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'AccessPolicyDTO'

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this AccessPolicyEntity. The URI for futures requests to the component.

**Returns** The uri of this AccessPolicyEntity.

**Return type** str

**nipyapi.nifi.models.access\_policy\_summary\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_policy_summary_dto.AccessPolicySummaryDTO (id=None,
                                                                              ver-
                                                                              sioned_component_id=
                                                                              par-
                                                                              ent_group_id=None,
                                                                              po-
                                                                              si-
                                                                              tion=None,
                                                                              re-
                                                                              source=None,
                                                                              ac-
                                                                              tion=None,
                                                                              com-
                                                                              po-
                                                                              nent_reference=None,
                                                                              con-
                                                                              fig-
                                                                              urable=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicySummaryDTO - a model defined in Swagger

**action**

Gets the action of this AccessPolicySummaryDTO. The action associated with this access policy.

**Returns** The action of this AccessPolicySummaryDTO.

**Return type** str

**attribute\_map** = {'action': 'action', 'component\_reference': 'componentReference', 'c

**component\_reference**

Gets the component\_reference of this AccessPolicySummaryDTO. Component this policy references if applicable.

**Returns** The component\_reference of this AccessPolicySummaryDTO.

**Return type** *ComponentReferenceEntity*

**configurable**

Gets the configurable of this AccessPolicySummaryDTO. Whether this policy is configurable.

**Returns** The configurable of this AccessPolicySummaryDTO.

**Return type** bool

**id**

Gets the id of this AccessPolicySummaryDTO. The id of the component.

**Returns** The id of this AccessPolicySummaryDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this AccessPolicySummaryDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this AccessPolicySummaryDTO.

**Return type** str

**position**

Gets the position of this AccessPolicySummaryDTO. The position of this component in the UI if applicable.

**Returns** The position of this AccessPolicySummaryDTO.

**Return type** *PositionDTO*

**resource**

Gets the resource of this AccessPolicySummaryDTO. The resource for this access policy.

**Returns** The resource of this AccessPolicySummaryDTO.

**Return type** str

**swagger\_types** = {'action': 'str', 'component\_reference': 'ComponentReferenceEntity',

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this AccessPolicySummaryDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this AccessPolicySummaryDTO.

**Return type** str

**nipyapi.nifi.models.access\_policy\_summary\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_policy_summary_entity.AccessPolicySummaryEntity (revision=None,
id=None,
uri=None,
position=None,
permissions=None,
bulletins=None,
disconnected_node_acknowledged=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicySummaryEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_node_acknowledged': 'disconnected_node_acknowledged'}
```

**bulletins**

Gets the bulletins of this AccessPolicySummaryEntity. The bulletins for this component.

**Returns** The bulletins of this AccessPolicySummaryEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this AccessPolicySummaryEntity.

**Returns** The component of this AccessPolicySummaryEntity.

**Return type** *AccessPolicySummaryDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this AccessPolicySummaryEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this AccessPolicySummaryEntity.

**Return type** bool

**id**

Gets the id of this AccessPolicySummaryEntity. The id of the component.

**Returns** The id of this AccessPolicySummaryEntity.

**Return type** str

**permissions**

Gets the permissions of this AccessPolicySummaryEntity. The permissions for this component.

**Returns** The permissions of this AccessPolicySummaryEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this AccessPolicySummaryEntity. The position of this component in the UI if applicable.

**Returns** The position of this AccessPolicySummaryEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this AccessPolicySummaryEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this AccessPolicySummaryEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'AccessPolicySummaryEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this AccessPolicySummaryEntity. The URI for futures requests to the component.

**Returns** The uri of this AccessPolicySummaryEntity.

**Return type** str

## nipyapi.nifi.models.access\_status\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_status_dto.AccessStatusDTO(identity=None,
                                                            status=None, message=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessStatusDTO - a model defined in Swagger

```
attribute_map = {'identity': 'identity', 'message': 'message', 'status': 'status'}
```

**identity**

Gets the identity of this AccessStatusDTO. The user identity.

**Returns** The identity of this AccessStatusDTO.

**Return type** str

**message**

Gets the message of this AccessStatusDTO. Additional details about the user access status.

**Returns** The message of this AccessStatusDTO.

**Return type** str

**status**

Gets the status of this AccessStatusDTO. The user access status.

**Returns** The status of this AccessStatusDTO.

**Return type** str

```
swagger_types = {'identity': 'str', 'message': 'str', 'status': 'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.access\_status\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.access_status_entity.AccessStatusEntity(access_status=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessStatusEntity - a model defined in Swagger

**access\_status**

Gets the access\_status of this AccessStatusEntity.

**Returns** The access\_status of this AccessStatusEntity.

**Return type** *AccessStatusDTO*

```
attribute_map = {'access_status': 'accessStatus'}
```

```
swagger_types = {'access_status': 'AccessStatusDTO'}
```

```
to_dict()  
    Returns the model properties as a dict  
  
to_str()  
    Returns the string representation of the model
```

### nipyapi.nifi.models.action\_details\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.action_details_dto.ActionDetailsDTO
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ActionDetailsDTO - a model defined in Swagger

```
attribute_map = {}  
swagger_types = {}  
  
to_dict()  
    Returns the model properties as a dict  
  
to_str()  
    Returns the string representation of the model
```

### nipyapi.nifi.models.action\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.action_dto.ActionDTO(id=None, user_identity=None,  
                                              timestamp=None, source_id=None,  
                                              source_name=None,  
                                              source_type=None, component=None,  
                                              nent_details=None, operation=None,  
                                              action_details=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ActionDTO - a model defined in Swagger

```
action_details  
    Gets the action_details of this ActionDTO. The details of the action.  
  
    Returns The action_details of this ActionDTO.
```



**Return type** *ActionDetailsDTO*

**attribute\_map** = {'action\_details': 'actionDetails', 'component\_details': 'componentD

**component\_details**

Gets the component\_details of this ActionDTO. The details of the source component.

**Returns** The component\_details of this ActionDTO.

**Return type** *ComponentDetailsDTO*

**id**

Gets the id of this ActionDTO. The action id.

**Returns** The id of this ActionDTO.

**Return type** int

**operation**

Gets the operation of this ActionDTO. The operation that was performed.

**Returns** The operation of this ActionDTO.

**Return type** str

**source\_id**

Gets the source\_id of this ActionDTO. The id of the source component.

**Returns** The source\_id of this ActionDTO.

**Return type** str

**source\_name**

Gets the source\_name of this ActionDTO. The name of the source component.

**Returns** The source\_name of this ActionDTO.

**Return type** str

**source\_type**

Gets the source\_type of this ActionDTO. The type of the source component.

**Returns** The source\_type of this ActionDTO.

**Return type** str

**swagger\_types** = {'action\_details': 'ActionDetailsDTO', 'component\_details': 'Componen

**timestamp**

Gets the timestamp of this ActionDTO. The timestamp of the action.

**Returns** The timestamp of this ActionDTO.

**Return type** str

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**user\_identity**

Gets the user\_identity of this ActionDTO. The identity of the user that performed the action.

**Returns** The user\_identity of this ActionDTO.

**Return type** str

## nipyapi.nifi.models.action\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.action_entity.ActionEntity(id=None, timestamp=None,
                                                    source_id=None,
                                                    can_read=None, action=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ActionEntity - a model defined in Swagger

#### **action**

Gets the action of this ActionEntity.

**Returns** The action of this ActionEntity.

**Return type** *ActionDTO*

```
attribute_map = {'action': 'action', 'can_read': 'canRead', 'id': 'id', 'source_id':
```

#### **can\_read**

Gets the can\_read of this ActionEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this ActionEntity.

**Return type** bool

#### **id**

Gets the id of this ActionEntity.

**Returns** The id of this ActionEntity.

**Return type** int

#### **source\_id**

Gets the source\_id of this ActionEntity.

**Returns** The source\_id of this ActionEntity.

**Return type** str

```
swagger_types = {'action': 'ActionDTO', 'can_read': 'bool', 'id': 'int', 'source_id':
```

#### **timestamp**

Gets the timestamp of this ActionEntity. The timestamp of the action.

**Returns** The timestamp of this ActionEntity.

**Return type** str

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.activate\_controller\_services\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.activate\_controller\_services\_entity.**ActivateControllerServicesEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ActivateControllerServicesEntity - a model defined in Swagger

**attribute\_map** = {'components': 'components', 'disconnected\_node\_acknowledged': 'disconnected\_node\_acknowledged'}

**components**

Gets the components of this ActivateControllerServicesEntity. Optional services to schedule. If not specified, all authorized descendant controller services will be used.

**Returns** The components of this ActivateControllerServicesEntity.

**Return type** dict(str, *RevisionDTO*)

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ActivateControllerServicesEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ActivateControllerServicesEntity.

**Return type** bool

**id**

Gets the id of this ActivateControllerServicesEntity. The id of the ProcessGroup

**Returns** The id of this ActivateControllerServicesEntity.

**Return type** str

**state**

Gets the state of this ActivateControllerServicesEntity. The desired state of the descendant components

**Returns** The state of this ActivateControllerServicesEntity.

**Return type** str

**swagger\_types** = {'components': 'dict(str, RevisionDTO)', 'disconnected\_node\_acknowledged': 'bool'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.affected\_component\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.affected_component_dto.AffectedComponentDTO(process_group_id=None,
                                                                    id=None,
                                                                    refer-
                                                                    ence_type=None,
                                                                    name=None,
                                                                    state=None,
                                                                    ac-
                                                                    tive_thread_count=None,
                                                                    val-
                                                                    ida-
                                                                    tion_errors=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AffectedComponentDTO - a model defined in Swagger

#### **active\_thread\_count**

Gets the active\_thread\_count of this AffectedComponentDTO. The number of active threads for the referencing component.

**Returns** The active\_thread\_count of this AffectedComponentDTO.

**Return type** int

```
attribute_map = {'active_thread_count': 'activeThreadCount', 'id': 'id', 'name': 'n
```

#### **id**

Gets the id of this AffectedComponentDTO. The UUID of this component

**Returns** The id of this AffectedComponentDTO.

**Return type** str

#### **name**

Gets the name of this AffectedComponentDTO. The name of this component.

**Returns** The name of this AffectedComponentDTO.

**Return type** str

#### **process\_group\_id**

Gets the process\_group\_id of this AffectedComponentDTO. The UUID of the Process Group that this component is in

**Returns** The process\_group\_id of this AffectedComponentDTO.

**Return type** str

#### **reference\_type**

Gets the reference\_type of this AffectedComponentDTO. The type of this component

**Returns** The reference\_type of this AffectedComponentDTO.

**Return type** str

**state**

Gets the state of this AffectedComponentDTO. The scheduled state of a processor or reporting task referencing a controller service. If this component is another controller service, this field represents the controller service state.

**Returns** The state of this AffectedComponentDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'id': 'str', 'name': 'str', 'process

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**validation\_errors**

Gets the validation\_errors of this AffectedComponentDTO. The validation errors for the component.

**Returns** The validation\_errors of this AffectedComponentDTO.

**Return type** list[str]

## nipyapi.nifi.models.affected\_component\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.affected_component_entity.AffectedComponentEntity(revision=None,
                                                                              id=None,
                                                                              uri=None,
                                                                              po-
                                                                              si-
                                                                              tion=None,
                                                                              per-
                                                                              mis-
                                                                              sions=None,
                                                                              bul-
                                                                              letins=None,
                                                                              dis-
                                                                              con-
                                                                              nected_node_acknow-
                                                                              com-
                                                                              po-
                                                                              nent=None,
                                                                              pro-
                                                                              cess_group=None,
                                                                              ref-
                                                                              er-
                                                                              ence_type=None)
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AffectedComponentEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'component': 'component', 'disconnected\_node\_

**bulletins**

Gets the bulletins of this AffectedComponentEntity. The bulletins for this component.

**Returns** The bulletins of this AffectedComponentEntity.

**Return type** list[[BulletinEntity](#)]

**component**

Gets the component of this AffectedComponentEntity.

**Returns** The component of this AffectedComponentEntity.

**Return type** [AffectedComponentDTO](#)

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this AffectedComponentEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this AffectedComponentEntity.

**Return type** bool

**id**

Gets the id of this AffectedComponentEntity. The id of the component.

**Returns** The id of this AffectedComponentEntity.

**Return type** str

**permissions**

Gets the permissions of this AffectedComponentEntity. The permissions for this component.

**Returns** The permissions of this AffectedComponentEntity.

**Return type** [PermissionsDTO](#)

**position**

Gets the position of this AffectedComponentEntity. The position of this component in the UI if applicable.

**Returns** The position of this AffectedComponentEntity.

**Return type** [PositionDTO](#)

**process\_group**

Gets the process\_group of this AffectedComponentEntity. The Process Group that the component belongs to

**Returns** The process\_group of this AffectedComponentEntity.

**Return type** ProcessGroupNameDTO

**reference\_type**

Gets the reference\_type of this AffectedComponentEntity. The type of component referenced

**Returns** The reference\_type of this AffectedComponentEntity.

**Return type** str

**revision**

Gets the revision of this AffectedComponentEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this AffectedComponentEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'AffectedComponentEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this AffectedComponentEntity. The URI for futures requests to the component.

**Returns** The uri of this AffectedComponentEntity.

**Return type** str

**nipyapi.nifi.models.allowable\_value\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.allowable_value_dto.AllowableValueDTO (display_name=None,  
value=None,  
description=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AllowableValueDTO - a model defined in Swagger

**attribute\_map** = {'description': 'description', 'display\_name': 'displayName', 'value': 'value'}

**description**

Gets the description of this AllowableValueDTO. A description for this allowable value.

**Returns** The description of this AllowableValueDTO.

**Return type** str

**display\_name**

Gets the display\_name of this AllowableValueDTO. A human readable value that is allowed for the property descriptor.

**Returns** The display\_name of this AllowableValueDTO.

**Return type** str

**swagger\_types** = {'description': 'str', 'display\_name': 'str', 'value': 'str'}

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

**value**  
Gets the value of this AllowableValueDTO. A value that is allowed for the property descriptor.

**Returns** The value of this AllowableValueDTO.

**Return type** str

## nipyapi.nifi.models.allowable\_value\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.allowable\_value\_entity.**AllowableValueEntity**(*allowable\_value=None, can\_read=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AllowableValueEntity - a model defined in Swagger

**allowable\_value**  
Gets the allowable\_value of this AllowableValueEntity.

**Returns** The allowable\_value of this AllowableValueEntity.

**Return type** *AllowableValueDTO*

**attribute\_map** = {'allowable\_value': 'allowableValue', 'can\_read': 'canRead'}

**can\_read**  
Gets the can\_read of this AllowableValueEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this AllowableValueEntity.

**Return type** bool

**swagger\_types** = {'allowable\_value': 'AllowableValueDTO', 'can\_read': 'bool'}

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

## nipyapi.nifi.models.attribute\_dto module

### NiFi Rest API



The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.attribute\_dto.**AttributeDTO** (*name=None, value=None, previous\_value=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AttributeDTO - a model defined in Swagger

**attribute\_map** = {'name': 'name', 'previous\_value': 'previousValue', 'value': 'value'}

**name**

Gets the name of this AttributeDTO. The attribute name.

**Returns** The name of this AttributeDTO.

**Return type** str

**previous\_value**

Gets the previous\_value of this AttributeDTO. The value of the attribute before the event took place.

**Returns** The previous\_value of this AttributeDTO.

**Return type** str

**swagger\_types** = {'name': 'str', 'previous\_value': 'str', 'value': 'str'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**value**

Gets the value of this AttributeDTO. The attribute value.

**Returns** The value of this AttributeDTO.

**Return type** str

## nipyapi.nifi.models.banner\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.banner\_dto.**BannerDTO** (*header\_text=None, footer\_text=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BannerDTO - a model defined in Swagger

**attribute\_map** = {'footer\_text': 'footerText', 'header\_text': 'headerText'}

**footer\_text**

Gets the footer\_text of this BannerDTO. The footer text.

**Returns** The footer\_text of this BannerDTO.

**Return type** str

**header\_text**

Gets the header\_text of this BannerDTO. The header text.

**Returns** The header\_text of this BannerDTO.

**Return type** str

**swagger\_types** = {'footer\_text': 'str', 'header\_text': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.banner\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.banner\_entity.**BannerEntity**(banners=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BannerEntity - a model defined in Swagger

**attribute\_map** = {'banners': 'banners'}

**banners**

Gets the banners of this BannerEntity.

**Returns** The banners of this BannerEntity.

**Return type** *BannerDTO*

**swagger\_types** = {'banners': 'BannerDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.batch\_settings\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO (count=None,  
size=None,  
duration=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BatchSettingsDTO - a model defined in Swagger

```
attribute_map = {'count': 'count', 'duration': 'duration', 'size': 'size'}
```

**count**

Gets the count of this BatchSettingsDTO. Preferred number of flow files to include in a transaction.

**Returns** The count of this BatchSettingsDTO.

**Return type** int

**duration**

Gets the duration of this BatchSettingsDTO. Preferred amount of time that a transaction should span.

**Returns** The duration of this BatchSettingsDTO.

**Return type** str

**size**

Gets the size of this BatchSettingsDTO. Preferred number of bytes to include in a transaction.

**Returns** The size of this BatchSettingsDTO.

**Return type** str

```
swagger_types = {'count': 'int', 'duration': 'str', 'size': 'str'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

### nipyapi.nifi.models.batch\_size module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.batch_size.BatchSize (count=None, size=None, duration=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BatchSize - a model defined in Swagger

```
attribute_map = {'count': 'count', 'duration': 'duration', 'size': 'size'}
```

**count**

Gets the count of this BatchSize. Preferred number of flow files to include in a transaction.

**Returns** The count of this BatchSize.

**Return type** int

**duration**

Gets the duration of this BatchSize. Preferred amount of time that a transaction should span.

**Returns** The duration of this BatchSize.

**Return type** str

**size**

Gets the size of this BatchSize. Preferred number of bytes to include in a transaction.

**Returns** The size of this BatchSize.

**Return type** str

```
swagger_types = {'count': 'int', 'duration': 'str', 'size': 'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bucket module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bucket.Bucket(link=None, identifier=None, name=None,
                                         created_timestamp=None, description=None,
                                         allow_bundle_redeploy=None,
                                         allow_public_read=None, permissions=None,
                                         revision=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Bucket - a model defined in Swagger

**allow\_bundle\_redeploy**

Gets the allow\_bundle\_redeploy of this Bucket. Indicates if this bucket allows the same version of an extension bundle to be redeployed and thus overwrite the existing artifact. By default this is false.

**Returns** The allow\_bundle\_redeploy of this Bucket.

**Return type** bool

**allow\_public\_read**

Gets the allow\_public\_read of this Bucket. Indicates if this bucket allows read access to unauthenticated anonymous users

**Returns** The allow\_public\_read of this Bucket.

**Return type** bool

**attribute\_map** = {'allow\_bundle\_redeploy': 'allowBundleRedeploy', 'allow\_public\_read':

**created\_timestamp**

Gets the created\_timestamp of this Bucket. The timestamp of when the bucket was first created. This is set by the server at creation time.

**Returns** The created\_timestamp of this Bucket.

**Return type** int

**description**

Gets the description of this Bucket. A description of the bucket.

**Returns** The description of this Bucket.

**Return type** str

**identifier**

Gets the identifier of this Bucket. An ID to uniquely identify this object.

**Returns** The identifier of this Bucket.

**Return type** str

**link**

Gets the link of this Bucket. An WebLink to this entity.

**Returns** The link of this Bucket.

**Return type** JaxbLink

**name**

Gets the name of this Bucket. The name of the bucket.

**Returns** The name of this Bucket.

**Return type** str

**permissions**

Gets the permissions of this Bucket. The access that the current user has to this bucket.

**Returns** The permissions of this Bucket.

**Return type** *Permissions*

**revision**

Gets the revision of this Bucket. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this Bucket.

**Return type** RevisionInfo

**swagger\_types** = {'allow\_bundle\_redeploy': 'bool', 'allow\_public\_read': 'bool', 'crea

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bucket\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bucket_dto.BucketDTO (id=None, name=None, description=None, created=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BucketDTO - a model defined in Swagger

```
attribute_map = {'created': 'created', 'description': 'description', 'id': 'id', 'name': 'name'}
```

#### **created**

Gets the created of this BucketDTO. The created timestamp of this bucket

**Returns** The created of this BucketDTO.

**Return type** int

#### **description**

Gets the description of this BucketDTO. The bucket description

**Returns** The description of this BucketDTO.

**Return type** str

#### **id**

Gets the id of this BucketDTO. The bucket identifier

**Returns** The id of this BucketDTO.

**Return type** str

#### **name**

Gets the name of this BucketDTO. The bucket name

**Returns** The name of this BucketDTO.

**Return type** str

```
swagger_types = {'created': 'int', 'description': 'str', 'id': 'str', 'name': 'str'}
```

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bucket\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.bucket\_entity.**BucketEntity** (*id=None, bucket=None, permissions=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BucketEntity - a model defined in Swagger

**attribute\_map** = {'bucket': 'bucket', 'id': 'id', 'permissions': 'permissions'}

**bucket**

Gets the bucket of this BucketEntity.

**Returns** The bucket of this BucketEntity.

**Return type** *BucketDTO*

**id**

Gets the id of this BucketEntity.

**Returns** The id of this BucketEntity.

**Return type** str

**permissions**

Gets the permissions of this BucketEntity.

**Returns** The permissions of this BucketEntity.

**Return type** *PermissionsDTO*

**swagger\_types** = {'bucket': 'BucketDTO', 'id': 'str', 'permissions': 'PermissionsDTO'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.buckets\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.buckets\_entity.**BucketsEntity** (*buckets=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BucketsEntity - a model defined in Swagger

**attribute\_map** = {'buckets': 'buckets'}

**buckets**

Gets the buckets of this BucketsEntity.

**Returns** The buckets of this BucketsEntity.

**Return type** list[*BucketEntity*]

**swagger\_types** = {'buckets': 'list[BucketEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bulletin\_board\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bulletin_board_dto.BulletinBoardDTO (bulletins=None,  
                                                             gener-  
                                                             ated=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BulletinBoardDTO - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'generated': 'generated'}

**bulletins**

Gets the bulletins of this BulletinBoardDTO. The bulletins in the bulletin board, that matches the supplied request.

**Returns** The bulletins of this BulletinBoardDTO.

**Return type** list[*BulletinEntity*]

**generated**

Gets the generated of this BulletinBoardDTO. The timestamp when this report was generated.

**Returns** The generated of this BulletinBoardDTO.

**Return type** str

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'generated': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bulletin\_board\_entity module

### NiFi Rest API



The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bulletin_board_entity.BulletinBoardEntity(bulletin_board=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BulletinBoardEntity - a model defined in Swagger

```
attribute_map = {'bulletin_board': 'bulletinBoard'}
```

```
bulletin_board
```

Gets the bulletin\_board of this BulletinBoardEntity.

**Returns** The bulletin\_board of this BulletinBoardEntity.

**Return type** *BulletinBoardDTO*

```
swagger_types = {'bulletin_board': 'BulletinBoardDTO'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

### nipyapi.nifi.models.bulletin\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bulletin_dto.BulletinDTO(id=None, node_address=None,
                                                    category=None, group_id=None,
                                                    source_id=None,
                                                    source_name=None, level=None,
                                                    message=None, times-
                                                    tamp=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BulletinDTO - a model defined in Swagger

```
attribute_map = {'category': 'category', 'group_id': 'groupId', 'id': 'id', 'level':
```

```
category
```

Gets the category of this BulletinDTO. The category of this bulletin.

**Returns** The category of this BulletinDTO.

**Return type** str

**group\_id**

Gets the group\_id of this BulletinDTO. The group id of the source component.

**Returns** The group\_id of this BulletinDTO.

**Return type** str

**id**

Gets the id of this BulletinDTO. The id of the bulletin.

**Returns** The id of this BulletinDTO.

**Return type** int

**level**

Gets the level of this BulletinDTO. The level of the bulletin.

**Returns** The level of this BulletinDTO.

**Return type** str

**message**

Gets the message of this BulletinDTO. The bulletin message.

**Returns** The message of this BulletinDTO.

**Return type** str

**node\_address**

Gets the node\_address of this BulletinDTO. If clustered, the address of the node from which the bulletin originated.

**Returns** The node\_address of this BulletinDTO.

**Return type** str

**source\_id**

Gets the source\_id of this BulletinDTO. The id of the source component.

**Returns** The source\_id of this BulletinDTO.

**Return type** str

**source\_name**

Gets the source\_name of this BulletinDTO. The name of the source component.

**Returns** The source\_name of this BulletinDTO.

**Return type** str

**swagger\_types** = {'category': 'str', 'group\_id': 'str', 'id': 'int', 'level': 'str'}

**timestamp**

Gets the timestamp of this BulletinDTO. When this bulletin was generated.

**Returns** The timestamp of this BulletinDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.bulletin\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bulletin_entity.BulletinEntity(id=None,
                                                         group_id=None,
                                                         source_id=None,
                                                         timestamp=None,
                                                         node_address=None,
                                                         can_read=None,    bul-
                                                         letin=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BulletinEntity - a model defined in Swagger

```
attribute_map = {'bulletin':  'bulletin', 'can_read':  'canRead', 'group_id':  'groupI
```

#### bulletin

Gets the bulletin of this BulletinEntity.

**Returns** The bulletin of this BulletinEntity.

**Return type** *BulletinDTO*

#### can\_read

Gets the can\_read of this BulletinEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this BulletinEntity.

**Return type** bool

#### group\_id

Gets the group\_id of this BulletinEntity.

**Returns** The group\_id of this BulletinEntity.

**Return type** str

#### id

Gets the id of this BulletinEntity.

**Returns** The id of this BulletinEntity.

**Return type** int

#### node\_address

Gets the node\_address of this BulletinEntity.

**Returns** The node\_address of this BulletinEntity.

**Return type** str

#### source\_id

Gets the source\_id of this BulletinEntity.

**Returns** The source\_id of this BulletinEntity.

**Return type** str

```
swagger_types = {'bulletin': 'BulletinDTO', 'can_read': 'bool', 'group_id': 'str',
timestamp
```

Gets the timestamp of this BulletinEntity. When this bulletin was generated.

**Returns** The timestamp of this BulletinEntity.

**Return type** str

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.bundle module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.bundle.Bundle(group=None, artifact=None, version=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Bundle - a model defined in Swagger

```
artifact
```

Gets the artifact of this Bundle. The artifact of the bundle

**Returns** The artifact of this Bundle.

**Return type** str

```
attribute_map = {'artifact': 'artifact', 'group': 'group', 'version': 'version'}
```

```
group
```

Gets the group of this Bundle. The group of the bundle

**Returns** The group of this Bundle.

**Return type** str

```
swagger_types = {'artifact': 'str', 'group': 'str', 'version': 'str'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

```
version
```

Gets the version of this Bundle. The version of the bundle

**Returns** The version of this Bundle.

**Return type** str

## nipyapi.nifi.models.bundle\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.bundle\_dto.**BundleDTO** (*group=None, artifact=None, version=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BundleDTO - a model defined in Swagger

#### **artifact**

Gets the artifact of this BundleDTO. The artifact of the bundle.

**Returns** The artifact of this BundleDTO.

**Return type** str

**attribute\_map** = {'artifact': 'artifact', 'group': 'group', 'version': 'version'}

#### **group**

Gets the group of this BundleDTO. The group of the bundle.

**Returns** The group of this BundleDTO.

**Return type** str

**swagger\_types** = {'artifact': 'str', 'group': 'str', 'version': 'str'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **version**

Gets the version of this BundleDTO. The version of the bundle.

**Returns** The version of this BundleDTO.

**Return type** str

## nipyapi.nifi.models.cluste\_summary\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.cluste_summary_entity.ClusteSummaryEntity(cluster_summary=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    ClusteSummaryEntity - a model defined in Swagger

    attribute_map = {'cluster_summary': 'clusterSummary'}

    cluster_summary
        Gets the cluster_summary of this ClusteSummaryEntity.

        Returns The cluster_summary of this ClusteSummaryEntity.

        Return type ClusterSummaryDTO

    swagger_types = {'cluster_summary': 'ClusterSummaryDTO'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model
```

## nipyapi.nifi.models.cluster\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.cluster_dto.ClusterDTO(nodes=None, generated=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    ClusterDTO - a model defined in Swagger

    attribute_map = {'generated': 'generated', 'nodes': 'nodes'}

    generated
        Gets the generated of this ClusterDTO. The timestamp the report was generated.

        Returns The generated of this ClusterDTO.

        Return type str

    nodes
        Gets the nodes of this ClusterDTO. The collection of nodes that are part of the cluster.

        Returns The nodes of this ClusterDTO.

        Return type list[NodeDTO]

    swagger_types = {'generated': 'str', 'nodes': 'list[NodeDTO]'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model
```

## nipyapi.nifi.models.cluster\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.cluster\_entity.ClusterEntity (*cluster=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ClusterEntity - a model defined in Swagger

**attribute\_map** = {'cluster': 'cluster'}

**cluster**

Gets the cluster of this ClusterEntity.

**Returns** The cluster of this ClusterEntity.

**Return type** ClusterDTO

**swagger\_types** = {'cluster': 'ClusterDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.cluster\_search\_results\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.cluster\_search\_results\_entity.ClusterSearchResultsEntity (*node\_results*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ClusterSearchResultsEntity - a model defined in Swagger

**attribute\_map** = {'node\_results': 'nodeResults'}

**node\_results**

Gets the node\_results of this ClusterSearchResultsEntity.

**Returns** The node\_results of this ClusterSearchResultsEntity.

**Return type** list[NodeSearchResultDTO]

**swagger\_types** = {'node\_results': 'list[NodeSearchResultDTO]'}

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

## nipyapi.nifi.models.cluster\_summary\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO(connected_nodes=None,  
                                                                con-  
                                                                nected_node_count=None,  
                                                                to-  
                                                                tal_node_count=None,  
                                                                clus-  
                                                                tered=None,  
                                                                con-  
                                                                nected_to_cluster=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ClusterSummaryDTO - a model defined in Swagger

**attribute\_map** = {'clustered': 'clustered', 'connected\_node\_count': 'connectedNodeCount'}

#### **clustered**

Gets the clustered of this ClusterSummaryDTO. Whether this NiFi instance is clustered.

**Returns** The clustered of this ClusterSummaryDTO.

**Return type** bool

#### **connected\_node\_count**

Gets the connected\_node\_count of this ClusterSummaryDTO. The number of nodes that are currently connected to the cluster

**Returns** The connected\_node\_count of this ClusterSummaryDTO.

**Return type** int

#### **connected\_nodes**

Gets the connected\_nodes of this ClusterSummaryDTO. When clustered, reports the number of nodes connected vs the number of nodes in the cluster.

**Returns** The connected\_nodes of this ClusterSummaryDTO.

**Return type** str

#### **connected\_to\_cluster**

Gets the connected\_to\_cluster of this ClusterSummaryDTO. Whether this NiFi instance is connected to a cluster.

**Returns** The connected\_to\_cluster of this ClusterSummaryDTO.



**Return type** bool

**swagger\_types** = {'clustered': 'bool', 'connected\_node\_count': 'int', 'connected\_node

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**total\_node\_count**

Gets the total\_node\_count of this ClusterSummaryDTO. The number of nodes in the cluster, regardless of whether or not they are connected

**Returns** The total\_node\_count of this ClusterSummaryDTO.

**Return type** int

### nipyapi.nifi.models.component\_details\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.component\_details\_dto.**ComponentDetailsDTO**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentDetailsDTO - a model defined in Swagger

**attribute\_map** = {}

**swagger\_types** = {}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

### nipyapi.nifi.models.component\_difference\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_difference_dto.ComponentDifferenceDTO(component_type=None,  
                                                                           com-  
                                                                           po-  
                                                                           nent_id=None,  
                                                                           com-  
                                                                           po-  
                                                                           nent_name=None,  
                                                                           pro-  
                                                                           cess_group_id=None,  
                                                                           dif-  
                                                                           fer-  
                                                                           ences=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentDifferenceDTO - a model defined in Swagger

```
attribute_map = {'component_id': 'componentId', 'component_name': 'componentName', 'process_group_id': 'processGroupId', 'differences': 'differences'}
```

**component\_id**

Gets the component\_id of this ComponentDifferenceDTO. The ID of the component

**Returns** The component\_id of this ComponentDifferenceDTO.

**Return type** str

**component\_name**

Gets the component\_name of this ComponentDifferenceDTO. The name of the component

**Returns** The component\_name of this ComponentDifferenceDTO.

**Return type** str

**component\_type**

Gets the component\_type of this ComponentDifferenceDTO. The type of component

**Returns** The component\_type of this ComponentDifferenceDTO.

**Return type** str

**differences**

Gets the differences of this ComponentDifferenceDTO. The differences in the component between the two flows

**Returns** The differences of this ComponentDifferenceDTO.

**Return type** list[[DifferenceDTO](#)]

**process\_group\_id**

Gets the process\_group\_id of this ComponentDifferenceDTO. The ID of the Process Group that the component belongs to

**Returns** The process\_group\_id of this ComponentDifferenceDTO.

**Return type** str

```
swagger_types = {'component_id': 'str', 'component_name': 'str', 'component_type': 'str', 'process_group_id': 'str', 'differences': 'list[DifferenceDTO]'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.component\_history\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_history_dto.ComponentHistoryDTO(component_id=None,
                                                                    prop-
                                                                    erty_history=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentHistoryDTO - a model defined in Swagger

```
attribute_map = {'component_id':  'componentId', 'property_history':  'propertyHistory'}
component_id
```

Gets the component\_id of this ComponentHistoryDTO. The component id.

**Returns** The component\_id of this ComponentHistoryDTO.

**Return type** str

```
property_history
```

Gets the property\_history of this ComponentHistoryDTO. The history for the properties of the component.

**Returns** The property\_history of this ComponentHistoryDTO.

**Return type** dict(str, *PropertyHistoryDTO*)

```
swagger_types = {'component_id':  'str', 'property_history':  'dict(str, PropertyHistoryDTO)'}
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.component\_history\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_history_entity.ComponentHistoryEntity(component_history=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentHistoryEntity - a model defined in Swagger

```
attribute_map = {'component_history':  'componentHistory'}
```

**component\_history**

Gets the component\_history of this ComponentHistoryEntity.

**Returns** The component\_history of this ComponentHistoryEntity.

**Return type** *ComponentHistoryDTO*

**swagger\_types** = {'component\_history': 'ComponentHistoryDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.component\_reference\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_reference_dto.ComponentReferenceDTO (id=None,
                                                                    ver-
                                                                    sioned_component_id=None,
                                                                    par-
                                                                    ent_group_id=None,
                                                                    po-
                                                                    si-
                                                                    tion=None,
                                                                    name=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentReferenceDTO - a model defined in Swagger

**attribute\_map** = {'id': 'id', 'name': 'name', 'parent\_group\_id': 'parentGroupId', 'p

**id**

Gets the id of this ComponentReferenceDTO. The id of the component.

**Returns** The id of this ComponentReferenceDTO.

**Return type** str

**name**

Gets the name of this ComponentReferenceDTO. The name of the component.

**Returns** The name of this ComponentReferenceDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ComponentReferenceDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ComponentReferenceDTO.

**Return type** str

**position**

Gets the position of this ComponentReferenceDTO. The position of this component in the UI if applicable.

**Returns** The position of this ComponentReferenceDTO.

**Return type** *PositionDTO*

**swagger\_types** = {'id': 'str', 'name': 'str', 'parent\_group\_id': 'str', 'position':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this ComponentReferenceDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ComponentReferenceDTO.

**Return type** str

## nipyapi.nifi.models.component\_reference\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity(revision=None,
                                                                              id=None,
                                                                              uri=None,
                                                                              position=None,
                                                                              permissions=None,
                                                                              bulletins=None,
                                                                              disconnected_node_acknowledgments=None,
                                                                              parent_group_id=None,
                                                                              component_id=None,
                                                                              position=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentReferenceEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this ComponentReferenceEntity. The bulletins for this component.

**Returns** The bulletins of this ComponentReferenceEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this ComponentReferenceEntity.

**Returns** The component of this ComponentReferenceEntity.

**Return type** *ComponentReferenceDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ComponentReferenceEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ComponentReferenceEntity.

**Return type** bool

**id**

Gets the id of this ComponentReferenceEntity. The id of the component.

**Returns** The id of this ComponentReferenceEntity.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ComponentReferenceEntity. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ComponentReferenceEntity.

**Return type** str

**permissions**

Gets the permissions of this ComponentReferenceEntity. The permissions for this component.

**Returns** The permissions of this ComponentReferenceEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this ComponentReferenceEntity. The position of this component in the UI if applicable.

**Returns** The position of this ComponentReferenceEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this ComponentReferenceEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ComponentReferenceEntity.

**Return type** *RevisionDTO*

```
swagger_types = {'bulletins': 'list[BulletinEntity]', 'component': 'ComponentReferen
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this ComponentReferenceEntity. The URI for futures requests to the component.

**Returns** The uri of this ComponentReferenceEntity.

**Return type** str

## nipyapi.nifi.models.component\_search\_result\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_search_result_dto.ComponentSearchResultDTO (id=None,
                                                                    group_id=None,
                                                                    parent_group=None,
                                                                    versioned_group=None,
                                                                    name=None,
                                                                    matches=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentSearchResultDTO - a model defined in Swagger

```
attribute_map = {'group_id': 'groupId', 'id': 'id', 'matches': 'matches', 'name':
```

**group\_id**

Gets the group\_id of this ComponentSearchResultDTO. The group id of the component that matched the search.

**Returns** The group\_id of this ComponentSearchResultDTO.

**Return type** str

**id**

Gets the id of this ComponentSearchResultDTO. The id of the component that matched the search.

**Returns** The id of this ComponentSearchResultDTO.

**Return type** str

**matches**

Gets the matches of this ComponentSearchResultDTO. What matched the search from the component.

**Returns** The matches of this ComponentSearchResultDTO.

**Return type** list[str]

**name**

Gets the name of this ComponentSearchResultDTO. The name of the component that matched the search.

**Returns** The name of this ComponentSearchResultDTO.

**Return type** str

**parent\_group**

Gets the parent\_group of this ComponentSearchResultDTO. The parent group of the component that matched the search.

**Returns** The parent\_group of this ComponentSearchResultDTO.

**Return type** SearchResultGroupDTO

**swagger\_types** = {'group\_id': 'str', 'id': 'str', 'matches': 'list[str]', 'name': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_group**

Gets the versioned\_group of this ComponentSearchResultDTO. The nearest versioned ancestor group of the component that matched the search.

**Returns** The versioned\_group of this ComponentSearchResultDTO.

**Return type** SearchResultGroupDTO

## nipyapi.nifi.models.component\_state\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.component_state_dto.ComponentStateDTO(component_id=None,  
                                                                state_description=None,  
                                                                cluster_state=None,  
                                                                local_state=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentStateDTO - a model defined in Swagger

**attribute\_map** = {'cluster\_state': 'clusterState', 'component\_id': 'componentId', 'local\_state': 'localState'}

**cluster\_state**

Gets the cluster\_state of this ComponentStateDTO. The cluster state for this component, or null if this NiFi is a standalone instance.

**Returns** The cluster\_state of this ComponentStateDTO.

**Return type** *StateMapDTO*

**component\_id**

Gets the component\_id of this ComponentStateDTO. The component identifier.

**Returns** The component\_id of this ComponentStateDTO.



**Return type** str

**local\_state**

Gets the local\_state of this ComponentStateDTO. The local state for this component.

**Returns** The local\_state of this ComponentStateDTO.

**Return type** *StateMapDTO*

**state\_description**

Gets the state\_description of this ComponentStateDTO. Description of the state this component persists.

**Returns** The state\_description of this ComponentStateDTO.

**Return type** str

**swagger\_types** = {'cluster\_state': 'StateMapDTO', 'component\_id': 'str', 'local\_state'

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.component\_state\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.component\_state\_entity.**ComponentStateEntity**(component\_state=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ComponentStateEntity - a model defined in Swagger

**attribute\_map** = {'component\_state': 'componentState'}

**component\_state**

Gets the component\_state of this ComponentStateEntity. The component state.

**Returns** The component\_state of this ComponentStateEntity.

**Return type** *ComponentStateDTO*

**swagger\_types** = {'component\_state': 'ComponentStateDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.connectable\_component module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connectable_component.ConnectableComponent (id=None,  
type=None,  
group_id=None,  
name=None,  
comments=None,  
instance_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectableComponent - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'group_id': 'groupId', 'id': 'id', 'instance_identifier': 'instanceIdentifier'}
```

#### comments

Gets the comments of this ConnectableComponent. The comments for the connectable component.

**Returns** The comments of this ConnectableComponent.

**Return type** str

#### group\_id

Gets the group\_id of this ConnectableComponent. The id of the group that the connectable component resides in

**Returns** The group\_id of this ConnectableComponent.

**Return type** str

#### id

Gets the id of this ConnectableComponent. The id of the connectable component.

**Returns** The id of this ConnectableComponent.

**Return type** str

#### instance\_identifier

Gets the instance\_identifier of this ConnectableComponent. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this ConnectableComponent.

**Return type** str

#### name

Gets the name of this ConnectableComponent. The name of the connectable component

**Returns** The name of this ConnectableComponent.

**Return type** str

```
swagger_types = {'comments': 'str', 'group_id': 'str', 'id': 'str', 'instance_identifier': 'str'}
```

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

**type**  
Gets the type of this ConnectableComponent. The type of component the connectable is.

**Returns** The type of this ConnectableComponent.

**Return type** str

### nipyapi.nifi.models.connectable\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connectable_dto.ConnectableDTO(id=None, versioned_component_id=None, type=None, group_id=None, name=None, running=None, transmitting=None, exists=None, comments=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectableDTO - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'exists': 'exists', 'group_id': 'groupId',
```

#### **comments**

Gets the comments of this ConnectableDTO. The comments for the connectable component.

**Returns** The comments of this ConnectableDTO.

**Return type** str

#### **exists**

Gets the exists of this ConnectableDTO. If the connectable component represents a remote port, indicates if the target exists.

**Returns** The exists of this ConnectableDTO.

**Return type** bool

#### **group\_id**

Gets the group\_id of this ConnectableDTO. The id of the group that the connectable component resides in

**Returns** The group\_id of this ConnectableDTO.

**Return type** str

**id**

Gets the id of this ConnectableDTO. The id of the connectable component.

**Returns** The id of this ConnectableDTO.

**Return type** str

**name**

Gets the name of this ConnectableDTO. The name of the connectable component

**Returns** The name of this ConnectableDTO.

**Return type** str

**running**

Gets the running of this ConnectableDTO. Reflects the current state of the connectable component.

**Returns** The running of this ConnectableDTO.

**Return type** bool

**swagger\_types** = {'comments': 'str', 'exists': 'bool', 'group\_id': 'str', 'id': 'st

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmitting**

Gets the transmitting of this ConnectableDTO. If the connectable component represents a remote port, indicates if the target is configured to transmit.

**Returns** The transmitting of this ConnectableDTO.

**Return type** bool

**type**

Gets the type of this ConnectableDTO. The type of component the connectable is.

**Returns** The type of this ConnectableDTO.

**Return type** str

**versioned\_component\_id**

Gets the versioned\_component\_id of this ConnectableDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ConnectableDTO.

**Return type** str

## nipyapi.nifi.models.connection\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.connection_dto.ConnectionDTO (id=None,
                                                         versioned_component_id=None,
                                                         parent_group_id=None, position=None, source=None,
                                                         destination=None,
                                                         name=None, label_index=None,
                                                         get_index=None, selected_relationships=None,
                                                         available_relationships=None,
                                                         back_pressure_object_threshold=None,
                                                         back_pressure_data_size_threshold=None,
                                                         flow_file_expiration=None,
                                                         prioritizers=None,
                                                         bends=None,
                                                         load_balance_strategy=None,
                                                         load_balance_partition_attribute=None,
                                                         load_balance_compression=None,
                                                         load_balance_status=None)

```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionDTO - a model defined in Swagger

```
attribute_map = {'available_relationships': 'availableRelationships', 'back_pressure_
```

#### **available\_relationships**

Gets the available\_relationships of this ConnectionDTO. The relationships that the source of the connection currently supports.

**Returns** The available\_relationships of this ConnectionDTO.

**Return type** list[str]

#### **back\_pressure\_data\_size\_threshold**

Gets the back\_pressure\_data\_size\_threshold of this ConnectionDTO. The object data size threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The back\_pressure\_data\_size\_threshold of this ConnectionDTO.

**Return type** str

#### **back\_pressure\_object\_threshold**

Gets the back\_pressure\_object\_threshold of this ConnectionDTO. The object count threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The back\_pressure\_object\_threshold of this ConnectionDTO.

**Return type** int

#### **bends**

Gets the bends of this ConnectionDTO. The bend points on the connection.

**Returns** The bends of this ConnectionDTO.

**Return type** list[[PositionDTO](#)]

**destination**

Gets the destination of this ConnectionDTO. The destination of the connection.

**Returns** The destination of this ConnectionDTO.

**Return type** *ConnectableDTO*

**flow\_file\_expiration**

Gets the flow\_file\_expiration of this ConnectionDTO. The amount of time a flow file may be in the flow before it will be automatically aged out of the flow. Once a flow file reaches this age it will be terminated from the flow the next time a processor attempts to start work on it.

**Returns** The flow\_file\_expiration of this ConnectionDTO.

**Return type** str

**getz\_index**

Gets the getz\_index of this ConnectionDTO. The z index of the connection.

**Returns** The getz\_index of this ConnectionDTO.

**Return type** int

**id**

Gets the id of this ConnectionDTO. The id of the component.

**Returns** The id of this ConnectionDTO.

**Return type** str

**label\_index**

Gets the label\_index of this ConnectionDTO. The index of the bend point where to place the connection label.

**Returns** The label\_index of this ConnectionDTO.

**Return type** int

**load\_balance\_compression**

Gets the load\_balance\_compression of this ConnectionDTO. Whether or not data should be compressed when being transferred between nodes in the cluster.

**Returns** The load\_balance\_compression of this ConnectionDTO.

**Return type** str

**load\_balance\_partition\_attribute**

Gets the load\_balance\_partition\_attribute of this ConnectionDTO. The FlowFile Attribute to use for determining which node a FlowFile will go to if the Load Balancing Strategy is set to PARTITION\_BY\_ATTRIBUTE

**Returns** The load\_balance\_partition\_attribute of this ConnectionDTO.

**Return type** str

**load\_balance\_status**

Gets the load\_balance\_status of this ConnectionDTO. The current status of the Connection's Load Balancing Activities. Status can indicate that Load Balancing is not configured for the connection, that Load Balancing is configured but inactive (not currently transferring data to another node), or that Load Balancing is configured and actively transferring data to another node.

**Returns** The load\_balance\_status of this ConnectionDTO.

**Return type** str

**load\_balance\_strategy**

Gets the load\_balance\_strategy of this ConnectionDTO. How to load balance the data in this Connection across the nodes in the cluster.

**Returns** The load\_balance\_strategy of this ConnectionDTO.

**Return type** str

**name**

Gets the name of this ConnectionDTO. The name of the connection.

**Returns** The name of this ConnectionDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ConnectionDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ConnectionDTO.

**Return type** str

**position**

Gets the position of this ConnectionDTO. The position of this component in the UI if applicable.

**Returns** The position of this ConnectionDTO.

**Return type** *PositionDTO*

**prioritizers**

Gets the prioritizers of this ConnectionDTO. The comparators used to prioritize the queue.

**Returns** The prioritizers of this ConnectionDTO.

**Return type** list[str]

**selected\_relationships**

Gets the selected\_relationships of this ConnectionDTO. The selected relationship that comprise the connection.

**Returns** The selected\_relationships of this ConnectionDTO.

**Return type** list[str]

**source**

Gets the source of this ConnectionDTO. The source of the connection.

**Returns** The source of this ConnectionDTO.

**Return type** *ConnectableDTO*

**swagger\_types** = {'available\_relationships': 'list[str]', 'back\_pressure\_data\_size\_threshold': 'int', 'parent\_group\_id': 'str', 'position': 'int', 'prioritizers': 'list[str]', 'selected\_relationships': 'list[str]', 'source': 'str', 'to\_dict': 'dict', 'to\_str': 'str', 'versioned\_component\_id': 'str', 'load\_balance\_strategy': 'str', 'name': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this ConnectionDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ConnectionDTO.

**Return type** str

## nipyapi.nifi.models.connection\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connection_entity.ConnectionEntity (revision=None,
                                                                id=None,
                                                                uri=None,      po-
                                                                sition=None,
                                                                permissions=None,
                                                                bulletins=None,
                                                                discon-
                                                                nected_node_acknowledged=None,
                                                                component=None,
                                                                status=None,
                                                                bends=None,   la-
                                                                bel_index=None,
                                                                getz_index=None,
                                                                source_id=None,
                                                                source_group_id=None,
                                                                source_type=None,
                                                                destina-
                                                                tion_id=None,
                                                                destina-
                                                                tion_group_id=None,
                                                                destina-
                                                                tion_type=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionEntity - a model defined in Swagger

```
attribute_map = {'bends':  'bends', 'bulletins':  'bulletins', 'component':  'component'}
```

#### **bends**

Gets the bends of this ConnectionEntity. The bend points on the connection.

**Returns** The bends of this ConnectionEntity.

**Return type** list[*PositionDTO*]

#### **bulletins**

Gets the bulletins of this ConnectionEntity. The bulletins for this component.

**Returns** The bulletins of this ConnectionEntity.

**Return type** list[*BulletinEntity*]

#### **component**

Gets the component of this ConnectionEntity.

**Returns** The component of this ConnectionEntity.

**Return type** *ConnectionDTO*



**destination\_group\_id**

Gets the destination\_group\_id of this ConnectionEntity. The identifier of the group of the destination of this connection.

**Returns** The destination\_group\_id of this ConnectionEntity.

**Return type** str

**destination\_id**

Gets the destination\_id of this ConnectionEntity. The identifier of the destination of this connection.

**Returns** The destination\_id of this ConnectionEntity.

**Return type** str

**destination\_type**

Gets the destination\_type of this ConnectionEntity. The type of component the destination connectable is.

**Returns** The destination\_type of this ConnectionEntity.

**Return type** str

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ConnectionEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ConnectionEntity.

**Return type** bool

**getz\_index**

Gets the getz\_index of this ConnectionEntity. The z index of the connection.

**Returns** The getz\_index of this ConnectionEntity.

**Return type** int

**id**

Gets the id of this ConnectionEntity. The id of the component.

**Returns** The id of this ConnectionEntity.

**Return type** str

**label\_index**

Gets the label\_index of this ConnectionEntity. The index of the bend point where to place the connection label.

**Returns** The label\_index of this ConnectionEntity.

**Return type** int

**permissions**

Gets the permissions of this ConnectionEntity. The permissions for this component.

**Returns** The permissions of this ConnectionEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this ConnectionEntity. The position of this component in the UI if applicable.

**Returns** The position of this ConnectionEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this ConnectionEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ConnectionEntity.

**Return type** *RevisionDTO*

**source\_group\_id**

Gets the source\_group\_id of this ConnectionEntity. The identifier of the group of the source of this connection.

**Returns** The source\_group\_id of this ConnectionEntity.

**Return type** str

**source\_id**

Gets the source\_id of this ConnectionEntity. The identifier of the source of this connection.

**Returns** The source\_id of this ConnectionEntity.

**Return type** str

**source\_type**

Gets the source\_type of this ConnectionEntity. The type of component the source connectable is.

**Returns** The source\_type of this ConnectionEntity.

**Return type** str

**status**

Gets the status of this ConnectionEntity. The status of the connection.

**Returns** The status of this ConnectionEntity.

**Return type** *ConnectionStatusDTO*

**swagger\_types** = {'bends': 'list[PositionDTO]', 'bulletins': 'list[BulletinEntity]',

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**uri**

Gets the uri of this ConnectionEntity. The URI for futures requests to the component.

**Returns** The uri of this ConnectionEntity.

**Return type** str

## nipyapi.nifi.models.connection\_status\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connection_status_dto.ConnectionStatusDTO (id=None,
                                                                    group_id=None,
                                                                    name=None,
                                                                    stats_last_refreshed=None,
                                                                    source_id=None,
                                                                    source_name=None,
                                                                    destination_id=None,
                                                                    destination_name=None,
                                                                    aggregate_snapshot=None,
                                                                    node_snapshots=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionStatusDTO - a model defined in Swagger

#### **aggregate\_snapshot**

Gets the aggregate\_snapshot of this ConnectionStatusDTO. The status snapshot that represents the aggregate stats of the cluster

**Returns** The aggregate\_snapshot of this ConnectionStatusDTO.

**Return type** *ConnectionStatusSnapshotDTO*

```
attribute_map = {'aggregate_snapshot': 'aggregateSnapshot', 'destination_id': 'destinationId'}
```

#### **destination\_id**

Gets the destination\_id of this ConnectionStatusDTO. The ID of the destination component

**Returns** The destination\_id of this ConnectionStatusDTO.

**Return type** str

#### **destination\_name**

Gets the destination\_name of this ConnectionStatusDTO. The name of the destination component

**Returns** The destination\_name of this ConnectionStatusDTO.

**Return type** str

#### **group\_id**

Gets the group\_id of this ConnectionStatusDTO. The ID of the Process Group that the connection belongs to

**Returns** The group\_id of this ConnectionStatusDTO.

**Return type** str

#### **id**

Gets the id of this ConnectionStatusDTO. The ID of the connection

**Returns** The id of this ConnectionStatusDTO.

**Return type** str

#### **name**

Gets the name of this ConnectionStatusDTO. The name of the connection

**Returns** The name of this ConnectionStatusDTO.

**Return type** str

**node\_snapshots**

Gets the node\_snapshots of this ConnectionStatusDTO. A list of status snapshots for each node

**Returns** The node\_snapshots of this ConnectionStatusDTO.

**Return type** list[*NodeConnectionStatusSnapshotDTO*]

**source\_id**

Gets the source\_id of this ConnectionStatusDTO. The ID of the source component

**Returns** The source\_id of this ConnectionStatusDTO.

**Return type** str

**source\_name**

Gets the source\_name of this ConnectionStatusDTO. The name of the source component

**Returns** The source\_name of this ConnectionStatusDTO.

**Return type** str

**stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this ConnectionStatusDTO. The timestamp of when the stats were last refreshed

**Returns** The stats\_last\_refreshed of this ConnectionStatusDTO.

**Return type** str

**swagger\_types** = {'aggregate\_snapshot': 'ConnectionStatusSnapshotDTO', 'destination\_id'

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.connection\_status\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connection_status_entity.ConnectionStatusEntity (connection_status=None, can_read=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionStatusEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'connection\_status': 'connectionStatus'}

**can\_read**

Gets the can\_read of this ConnectionStatusEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this ConnectionStatusEntity.

**Return type** bool

**connection\_status**

Gets the connection\_status of this ConnectionStatusEntity.

**Returns** The connection\_status of this ConnectionStatusEntity.

**Return type** *ConnectionStatusDTO*

```
swagger_types = {'can_read': 'bool', 'connection_status': 'ConnectionStatusDTO'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.connection\_status\_snapshot\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO (id=None, group_id=None, name=None, source_id=None, source_name=None, destination_id=None, destination_name=None, predicates=None, flow_files=None, bytes_in=None, in_put=None, flow_files_in_put=None, bytes_out=None, out_put=None, flow_files_out_put=None, bytes_queued=None, queued_bytes=None, queued_bytes_per_cent_usage=None, per_cent_usage=None, flow_files_per_cent_usage=None, flow_files_per_cent_usage_per_flow_file=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionStatusSnapshotDTO - a model defined in Swagger

```
attribute_map = {'bytes_in': 'bytesIn', 'bytes_out': 'bytesOut', 'bytes_queued': 'bytesQueued'}
```

#### bytes\_in

Gets the bytes\_in of this ConnectionStatusSnapshotDTO. The size of the FlowFiles that have come into the connection in the last 5 minutes.

**Returns** The bytes\_in of this ConnectionStatusSnapshotDTO.

**Return type** int

#### bytes\_out

Gets the bytes\_out of this ConnectionStatusSnapshotDTO. The number of bytes that have left the connection in the last 5 minutes.

**Returns** The bytes\_out of this ConnectionStatusSnapshotDTO.

**Return type** int

**bytes\_queued**

Gets the bytes\_queued of this ConnectionStatusSnapshotDTO. The size of the FlowFiles that are currently queued in the connection.

**Returns** The bytes\_queued of this ConnectionStatusSnapshotDTO.

**Return type** int

**destination\_id**

Gets the destination\_id of this ConnectionStatusSnapshotDTO. The id of the destination of the connection.

**Returns** The destination\_id of this ConnectionStatusSnapshotDTO.

**Return type** str

**destination\_name**

Gets the destination\_name of this ConnectionStatusSnapshotDTO. The name of the destination of the connection.

**Returns** The destination\_name of this ConnectionStatusSnapshotDTO.

**Return type** str

**flow\_file\_availability**

Gets the flow\_file\_availability of this ConnectionStatusSnapshotDTO. The availability of FlowFiles in this connection

**Returns** The flow\_file\_availability of this ConnectionStatusSnapshotDTO.

**Return type** str

**flow\_files\_in**

Gets the flow\_files\_in of this ConnectionStatusSnapshotDTO. The number of FlowFiles that have come into the connection in the last 5 minutes.

**Returns** The flow\_files\_in of this ConnectionStatusSnapshotDTO.

**Return type** int

**flow\_files\_out**

Gets the flow\_files\_out of this ConnectionStatusSnapshotDTO. The number of FlowFiles that have left the connection in the last 5 minutes.

**Returns** The flow\_files\_out of this ConnectionStatusSnapshotDTO.

**Return type** int

**flow\_files\_queued**

Gets the flow\_files\_queued of this ConnectionStatusSnapshotDTO. The number of FlowFiles that are currently queued in the connection.

**Returns** The flow\_files\_queued of this ConnectionStatusSnapshotDTO.

**Return type** int

**group\_id**

Gets the group\_id of this ConnectionStatusSnapshotDTO. The id of the process group the connection belongs to.

**Returns** The group\_id of this ConnectionStatusSnapshotDTO.

**Return type** str

**id**

Gets the id of this ConnectionStatusSnapshotDTO. The id of the connection.

**Returns** The id of this ConnectionStatusSnapshotDTO.

**Return type** str

**input**

Gets the input of this ConnectionStatusSnapshotDTO. The input count/size for the connection in the last 5 minutes, pretty printed.

**Returns** The input of this ConnectionStatusSnapshotDTO.

**Return type** str

**name**

Gets the name of this ConnectionStatusSnapshotDTO. The name of the connection.

**Returns** The name of this ConnectionStatusSnapshotDTO.

**Return type** str

**output**

Gets the output of this ConnectionStatusSnapshotDTO. The output count/size for the connection in the last 5 minutes, pretty printed.

**Returns** The output of this ConnectionStatusSnapshotDTO.

**Return type** str

**percent\_use\_bytes**

Gets the percent\_use\_bytes of this ConnectionStatusSnapshotDTO. Connection percent use regarding queued flow files size and backpressure threshold if configured.

**Returns** The percent\_use\_bytes of this ConnectionStatusSnapshotDTO.

**Return type** int

**percent\_use\_count**

Gets the percent\_use\_count of this ConnectionStatusSnapshotDTO. Connection percent use regarding queued flow files count and backpressure threshold if configured.

**Returns** The percent\_use\_count of this ConnectionStatusSnapshotDTO.

**Return type** int

**predictions**

Gets the predictions of this ConnectionStatusSnapshotDTO. Predictions, if available, for this connection (null if not available)

**Returns** The predictions of this ConnectionStatusSnapshotDTO.

**Return type** ConnectionStatusPredictionsSnapshotDTO

**queued**

Gets the queued of this ConnectionStatusSnapshotDTO. The total count and size of queued flowfiles formatted.

**Returns** The queued of this ConnectionStatusSnapshotDTO.

**Return type** str

**queued\_count**

Gets the queued\_count of this ConnectionStatusSnapshotDTO. The number of flowfiles that are queued, pretty printed.

**Returns** The queued\_count of this ConnectionStatusSnapshotDTO.

**Return type** str



**queued\_size**

Gets the `queued_size` of this `ConnectionStatusSnapshotDTO`. The total size of flowfiles that are queued formatted.

**Returns** The `queued_size` of this `ConnectionStatusSnapshotDTO`.

**Return type** str

**source\_id**

Gets the `source_id` of this `ConnectionStatusSnapshotDTO`. The id of the source of the connection.

**Returns** The `source_id` of this `ConnectionStatusSnapshotDTO`.

**Return type** str

**source\_name**

Gets the `source_name` of this `ConnectionStatusSnapshotDTO`. The name of the source of the connection.

**Returns** The `source_name` of this `ConnectionStatusSnapshotDTO`.

**Return type** str

**swagger\_types** = {'bytes\_in': 'int', 'bytes\_out': 'int', 'bytes\_queued': 'int', 'des'

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.connection\_status\_snapshot\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.connection\_status\_snapshot\_entity.**ConnectionStatusSnapshotEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionStatusSnapshotEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'connection\_status\_snapshot': 'connectionSta

**can\_read**

Gets the `can_read` of this `ConnectionStatusSnapshotEntity`. Indicates whether the user can read a given resource.

**Returns** The `can_read` of this `ConnectionStatusSnapshotEntity`.

**Return type** bool

**connection\_status\_snapshot**

Gets the `connection_status_snapshot` of this `ConnectionStatusSnapshotEntity`.

**Returns** The connection\_status\_snapshot of this ConnectionStatusSnapshotEntity.

**Return type** *ConnectionStatusSnapshotDTO*

**id**

Gets the id of this ConnectionStatusSnapshotEntity. The id of the connection.

**Returns** The id of this ConnectionStatusSnapshotEntity.

**Return type** str

**swagger\_types** = {'can\_read': 'bool', 'connection\_status\_snapshot': 'ConnectionStatusSnapshotDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.connections\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.connections\_entity.**ConnectionsEntity**(connections=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectionsEntity - a model defined in Swagger

**attribute\_map** = {'connections': 'connections'}

**connections**

Gets the connections of this ConnectionsEntity.

**Returns** The connections of this ConnectionsEntity.

**Return type** list[*ConnectionEntity*]

**swagger\_types** = {'connections': 'list[ConnectionEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.controller\_bulletins\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_bulletins\_entity.ControllerBulletinsEntity (bulletins=None

con-  
troller\_service\_  
re-  
port-  
ing\_task\_bullet

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerBulletinsEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'controller\_service\_bulletins': 'controller\_service\_bulletins', 'reporting\_task\_bulletins': 'reporting\_task\_bulletins'}

**bulletins**  
Gets the bulletins of this ControllerBulletinsEntity. System level bulletins to be reported to the user.

**Returns** The bulletins of this ControllerBulletinsEntity.

**Return type** list[BulletinEntity]

**controller\_service\_bulletins**

Gets the controller\_service\_bulletins of this ControllerBulletinsEntity. Controller service bulletins to be reported to the user.

**Returns** The controller\_service\_bulletins of this ControllerBulletinsEntity.

**Return type** list[BulletinEntity]

**reporting\_task\_bulletins**

Gets the reporting\_task\_bulletins of this ControllerBulletinsEntity. Reporting task bulletins to be reported to the user.

**Returns** The reporting\_task\_bulletins of this ControllerBulletinsEntity.

**Return type** list[BulletinEntity]

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'controller\_service\_bulletins': 'list[BulletinEntity]', 'reporting\_task\_bulletins': 'list[BulletinEntity]'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.controller\_configuration\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_configuration\_dto.ControllerConfigurationDTO (max\_timer\_event\_processors=100, max\_event\_processors=100)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerConfigurationDTO - a model defined in Swagger

```
attribute_map = {'max_event_driven_thread_count': 'maxEventDrivenThreadCount', 'max_t
```

```
max_event_driven_thread_count
```

Gets the max\_event\_driven\_thread\_count of this ControllerConfigurationDTO. The maximum number of event driven threads the NiFi has available.

**Returns** The max\_event\_driven\_thread\_count of this ControllerConfigurationDTO.

**Return type** int

```
max_timer_driven_thread_count
```

Gets the max\_timer\_driven\_thread\_count of this ControllerConfigurationDTO. The maximum number of timer driven threads the NiFi has available.

**Returns** The max\_timer\_driven\_thread\_count of this ControllerConfigurationDTO.

**Return type** int

```
swagger_types = {'max_event_driven_thread_count': 'int', 'max_timer_driven_thread_coun
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

## nipyapi.nifi.models.controller\_configuration\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_configuration_entity.ControllerConfigurationEntity (rev
```

per  
mis  
sion  
dis-  
com  
nec  
com  
po-  
nem

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerConfigurationEntity - a model defined in Swagger

```
attribute_map = {'component': 'component', 'disconnected_node_acknowledged': 'discon
```

```
component
```

Gets the component of this ControllerConfigurationEntity. The controller configuration.

**Returns** The component of this ControllerConfigurationEntity.

**Return type** *ControllerConfigurationDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ControllerConfigurationEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ControllerConfigurationEntity.

**Return type** bool

**permissions**

Gets the permissions of this ControllerConfigurationEntity. The permissions for this component.

**Returns** The permissions of this ControllerConfigurationEntity.

**Return type** *PermissionsDTO*

**revision**

Gets the revision of this ControllerConfigurationEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ControllerConfigurationEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'component': 'ControllerConfigurationDTO', 'disconnected\_node\_acknowledged': bool, 'permissions': 'PermissionsDTO', 'revision': 'RevisionDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.controller\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_dto.ControllerDTO (id=None,      name=None,
                                                    comments=None,      run-
                                                    ning_count=None,
                                                    stopped_count=None,
                                                    invalid_count=None,
                                                    disabled_count=None, ac-
                                                    tive_remote_port_count=None,
                                                    inac-
                                                    tive_remote_port_count=None,
                                                    input_port_count=None,
                                                    out-
                                                    put_port_count=None, re-
                                                    mote_site_listening_port=None,
                                                    re-
                                                    mote_site_http_listening_port=None,
                                                    site_to_site_secure=None,
                                                    instance_id=None,      in-
                                                    put_ports=None,      out-
                                                    put_ports=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerDTO - a model defined in Swagger

#### **active\_remote\_port\_count**

Gets the active\_remote\_port\_count of this ControllerDTO. The number of active remote ports contained in the NiFi.

**Returns** The active\_remote\_port\_count of this ControllerDTO.

**Return type** int

```
attribute_map = {'active_remote_port_count':  'activeRemotePortCount', 'comments':  'c
```

#### **comments**

Gets the comments of this ControllerDTO. The comments for the NiFi.

**Returns** The comments of this ControllerDTO.

**Return type** str

#### **disabled\_count**

Gets the disabled\_count of this ControllerDTO. The number of disabled components in the NiFi.

**Returns** The disabled\_count of this ControllerDTO.

**Return type** int

#### **id**

Gets the id of this ControllerDTO. The id of the NiFi.

**Returns** The id of this ControllerDTO.

**Return type** str

#### **inactive\_remote\_port\_count**

Gets the inactive\_remote\_port\_count of this ControllerDTO. The number of inactive remote ports contained in the NiFi.

**Returns** The inactive\_remote\_port\_count of this ControllerDTO.

**Return type** int

**input\_port\_count**

Gets the input\_port\_count of this ControllerDTO. The number of input ports contained in the NiFi.

**Returns** The input\_port\_count of this ControllerDTO.

**Return type** int

**input\_ports**

Gets the input\_ports of this ControllerDTO. The input ports available to send data to for the NiFi.

**Returns** The input\_ports of this ControllerDTO.

**Return type** list[[PortDTO](#)]

**instance\_id**

Gets the instance\_id of this ControllerDTO. If clustered, the id of the Cluster Manager, otherwise the id of the NiFi.

**Returns** The instance\_id of this ControllerDTO.

**Return type** str

**invalid\_count**

Gets the invalid\_count of this ControllerDTO. The number of invalid components in the NiFi.

**Returns** The invalid\_count of this ControllerDTO.

**Return type** int

**name**

Gets the name of this ControllerDTO. The name of the NiFi.

**Returns** The name of this ControllerDTO.

**Return type** str

**output\_port\_count**

Gets the output\_port\_count of this ControllerDTO. The number of output ports in the NiFi.

**Returns** The output\_port\_count of this ControllerDTO.

**Return type** int

**output\_ports**

Gets the output\_ports of this ControllerDTO. The output ports available to received data from the NiFi.

**Returns** The output\_ports of this ControllerDTO.

**Return type** list[[PortDTO](#)]

**remote\_site\_http\_listening\_port**

Gets the remote\_site\_http\_listening\_port of this ControllerDTO. The HTTP(S) Port on which this instance is listening for Remote Transfers of Flow Files. If this instance is not configured to receive Flow Files from remote instances, this will be null.

**Returns** The remote\_site\_http\_listening\_port of this ControllerDTO.

**Return type** int

**remote\_site\_listening\_port**

Gets the remote\_site\_listening\_port of this ControllerDTO. The Socket Port on which this instance is listening for Remote Transfers of Flow Files. If this instance is not configured to receive Flow Files from remote instances, this will be null.

**Returns** The remote\_site\_listening\_port of this ControllerDTO.

**Return type** int

**running\_count**

Gets the running\_count of this ControllerDTO. The number of running components in the NiFi.

**Returns** The running\_count of this ControllerDTO.

**Return type** int

**site\_to\_site\_secure**

Gets the site\_to\_site\_secure of this ControllerDTO. Indicates whether or not Site-to-Site communications with this instance is secure (2-way authentication).

**Returns** The site\_to\_site\_secure of this ControllerDTO.

**Return type** bool

**stopped\_count**

Gets the stopped\_count of this ControllerDTO. The number of stopped components in the NiFi.

**Returns** The stopped\_count of this ControllerDTO.

**Return type** int

**swagger\_types** = {'active\_remote\_port\_count': 'int', 'comments': 'str', 'disabled\_cou

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.controller\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_entity.**ControllerEntity**(controller=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerEntity - a model defined in Swagger

**attribute\_map** = {'controller': 'controller'}

**controller**

Gets the controller of this ControllerEntity.

**Returns** The controller of this ControllerEntity.

**Return type** *ControllerDTO*

**swagger\_types** = {'controller': 'ControllerDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model



## nipyapi.nifi.models.controller\_service\_api module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_service_api.ControllerServiceAPI (type=None,
                                                                    bun-
                                                                    dle=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceAPI - a model defined in Swagger

```
attribute_map = {'bundle':  'bundle', 'type':  'type'}
```

#### bundle

Gets the bundle of this ControllerServiceAPI. The details of the artifact that bundled this service interface.

**Returns** The bundle of this ControllerServiceAPI.

**Return type** *Bundle*

```
swagger_types = {'bundle':  'Bundle', 'type':  'str'}
```

#### to\_dict()

Returns the model properties as a dict

#### to\_str()

Returns the string representation of the model

#### type

Gets the type of this ControllerServiceAPI. The fully qualified name of the service interface.

**Returns** The type of this ControllerServiceAPI.

**Return type** str

## nipyapi.nifi.models.controller\_service\_api\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_service_api_dto.ControllerServiceApiDTO (type=None,
                                                                    bun-
                                                                    dle=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceApiDTO - a model defined in Swagger

```
attribute_map = {'bundle': 'bundle', 'type': 'type'}
```

**bundle**

Gets the bundle of this ControllerServiceApiDTO. The details of the artifact that bundled this service interface.

**Returns** The bundle of this ControllerServiceApiDTO.

**Return type** *BundleDTO*

```
swagger_types = {'bundle': 'BundleDTO', 'type': 'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ControllerServiceApiDTO. The fully qualified name of the service interface.

**Returns** The type of this ControllerServiceApiDTO.

**Return type** str

## nipyapi.nifi.models.controller\_service\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO (id=None,
                                                                    ver-
                                                                    sioned_component_id=None,
                                                                    par-
                                                                    ent_group_id=None,
                                                                    posi-
                                                                    tion=None,
                                                                    name=None,
                                                                    type=None,
                                                                    bun-
                                                                    dle=None,
                                                                    con-
                                                                    troller_service_apis=None,
                                                                    com-
                                                                    ments=None,
                                                                    state=None,
                                                                    per-
                                                                    sists_state=None,
                                                                    re-
                                                                    stricted=None,
                                                                    dep-
                                                                    re-
                                                                    cated=None,
                                                                    multi-
                                                                    ple_versions_available=None,
                                                                    sup-
                                                                    ports_sensitive_dynamic_prop-
                                                                    prop-
                                                                    er-
                                                                    ties=None,
                                                                    de-
                                                                    scrip-
                                                                    tors=None,
                                                                    sensi-
                                                                    tive_dynamic_property_names=None,
                                                                    cus-
                                                                    tom_ui_url=None,
                                                                    an-
                                                                    nota-
                                                                    tion_data=None,
                                                                    refer-
                                                                    enc-
                                                                    ing_components=None,
                                                                    val-
                                                                    ida-
                                                                    tion_errors=None,
                                                                    val-
                                                                    ida-
                                                                    tion_status=None,
                                                                    bul-
                                                                    letin_level=None,
                                                                    exten-
                                                                    sion_missing=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceDTO - a model defined in Swagger

**annotation\_data**

Gets the annotation\_data of this ControllerServiceDTO. The annotation for the controller service. This is how the custom UI relays configuration to the controller service.

**Returns** The annotation\_data of this ControllerServiceDTO.

**Return type** str

**attribute\_map** = {'annotation\_data': 'annotationData', 'bulletin\_level': 'bulletinLevel'}

**bulletin\_level**

Gets the bulletin\_level of this ControllerServiceDTO. The level at which the controller service will report bulletins.

**Returns** The bulletin\_level of this ControllerServiceDTO.

**Return type** str

**bundle**

Gets the bundle of this ControllerServiceDTO. The details of the artifact that bundled this processor type.

**Returns** The bundle of this ControllerServiceDTO.

**Return type** *BundleDTO*

**comments**

Gets the comments of this ControllerServiceDTO. The comments for the controller service.

**Returns** The comments of this ControllerServiceDTO.

**Return type** str

**controller\_service\_apis**

Gets the controller\_service\_apis of this ControllerServiceDTO. Lists the APIs this Controller Service implements.

**Returns** The controller\_service\_apis of this ControllerServiceDTO.

**Return type** list[*ControllerServiceApiDTO*]

**custom\_ui\_url**

Gets the custom\_ui\_url of this ControllerServiceDTO. The URL for the controller services custom configuration UI if applicable.

**Returns** The custom\_ui\_url of this ControllerServiceDTO.

**Return type** str

**deprecated**

Gets the deprecated of this ControllerServiceDTO. Whether the controller service has been deprecated.

**Returns** The deprecated of this ControllerServiceDTO.

**Return type** bool

**descriptors**

Gets the descriptors of this ControllerServiceDTO. The descriptors for the controller service properties.

**Returns** The descriptors of this ControllerServiceDTO.

**Return type** dict(str, *PropertyDescriptorDTO*)

**extension\_missing**

Gets the extension\_missing of this ControllerServiceDTO. Whether the underlying extension is missing.

**Returns** The extension\_missing of this ControllerServiceDTO.

**Return type** bool

**id**

Gets the id of this ControllerServiceDTO. The id of the component.

**Returns** The id of this ControllerServiceDTO.

**Return type** str

**multiple\_versions\_available**

Gets the multiple\_versions\_available of this ControllerServiceDTO. Whether the controller service has multiple versions available.

**Returns** The multiple\_versions\_available of this ControllerServiceDTO.

**Return type** bool

**name**

Gets the name of this ControllerServiceDTO. The name of the controller service.

**Returns** The name of this ControllerServiceDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ControllerServiceDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ControllerServiceDTO.

**Return type** str

**persists\_state**

Gets the persists\_state of this ControllerServiceDTO. Whether the controller service persists state.

**Returns** The persists\_state of this ControllerServiceDTO.

**Return type** bool

**position**

Gets the position of this ControllerServiceDTO. The position of this component in the UI if applicable.

**Returns** The position of this ControllerServiceDTO.

**Return type** *PositionDTO*

**properties**

Gets the properties of this ControllerServiceDTO. The properties of the controller service.

**Returns** The properties of this ControllerServiceDTO.

**Return type** dict(str, str)

**referencing\_components**

Gets the referencing\_components of this ControllerServiceDTO. All components referencing this controller service.

**Returns** The referencing\_components of this ControllerServiceDTO.

**Return type** list[*ControllerServiceReferencingComponentEntity*]

**restricted**

Gets the restricted of this ControllerServiceDTO. Whether the controller service requires elevated privileges.

**Returns** The restricted of this ControllerServiceDTO.

**Return type** bool

**sensitive\_dynamic\_property\_names**

Gets the sensitive\_dynamic\_property\_names of this ControllerServiceDTO. Set of sensitive dynamic property names

**Returns** The sensitive\_dynamic\_property\_names of this ControllerServiceDTO.

**Return type** list[str]

**state**

Gets the state of this ControllerServiceDTO. The state of the controller service.

**Returns** The state of this ControllerServiceDTO.

**Return type** str

**supports\_sensitive\_dynamic\_properties**

Gets the supports\_sensitive\_dynamic\_properties of this ControllerServiceDTO. Whether the controller service supports sensitive dynamic properties.

**Returns** The supports\_sensitive\_dynamic\_properties of this ControllerServiceDTO.

**Return type** bool

**swagger\_types** = {'annotation\_data': 'str', 'bulletin\_level': 'str', 'bundle': 'Bund

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ControllerServiceDTO. The type of the controller service.

**Returns** The type of this ControllerServiceDTO.

**Return type** str

**validation\_errors**

Gets the validation\_errors of this ControllerServiceDTO. The validation errors from the controller service. These validation errors represent the problems with the controller service that must be resolved before it can be enabled.

**Returns** The validation\_errors of this ControllerServiceDTO.

**Return type** list[str]

**validation\_status**

Gets the validation\_status of this ControllerServiceDTO. Indicates whether the ControllerService is valid, invalid, or still in the process of validating (i.e., it is unknown whether or not the ControllerService is valid)

**Returns** The validation\_status of this ControllerServiceDTO.

**Return type** str

**versioned\_component\_id**

Gets the versioned\_component\_id of this ControllerServiceDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ControllerServiceDTO.

**Return type** str

## nipyapi.nifi.models.controller\_service\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity (revision=None,
                                                                    id=None,
                                                                    uri=None,
                                                                    po-
                                                                    si-
                                                                    tion=None,
                                                                    per-
                                                                    mis-
                                                                    sions=None,
                                                                    bul-
                                                                    letins=None,
                                                                    dis-
                                                                    con-
                                                                    nected_node_acknow-
                                                                    par-
                                                                    ent_group_id=None,
                                                                    com-
                                                                    po-
                                                                    nent=None,
                                                                    op-
                                                                    er-
                                                                    ate_permissions=None,
                                                                    sta-
                                                                    tus=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_n
```

#### bulletins

Gets the bulletins of this ControllerServiceEntity. The bulletins for this component.

**Returns** The bulletins of this ControllerServiceEntity.

**Return type** list[*BulletinEntity*]

#### component

Gets the component of this ControllerServiceEntity.

**Returns** The component of this ControllerServiceEntity.

**Return type** *ControllerServiceDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ControllerServiceEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ControllerServiceEntity.

**Return type** bool

**id**

Gets the id of this ControllerServiceEntity. The id of the component.

**Returns** The id of this ControllerServiceEntity.

**Return type** str

**operate\_permissions**

Gets the operate\_permissions of this ControllerServiceEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this ControllerServiceEntity.

**Return type** *PermissionsDTO*

**parent\_group\_id**

Gets the parent\_group\_id of this ControllerServiceEntity. The id of parent process group of this ControllerService.

**Returns** The parent\_group\_id of this ControllerServiceEntity.

**Return type** str

**permissions**

Gets the permissions of this ControllerServiceEntity. The permissions for this component.

**Returns** The permissions of this ControllerServiceEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this ControllerServiceEntity. The position of this component in the UI if applicable.

**Returns** The position of this ControllerServiceEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this ControllerServiceEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ControllerServiceEntity.

**Return type** *RevisionDTO*

**status**

Gets the status of this ControllerServiceEntity. The status for this ControllerService.

**Returns** The status of this ControllerServiceEntity.

**Return type** ControllerServiceStatusDTO

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'ControllerServiceEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model



**uri**

Gets the uri of this ControllerServiceEntity. The URI for futures requests to the component.

**Returns** The uri of this ControllerServiceEntity.

**Return type** str

### nipyapi.nifi.models.controller\_service\_referencing\_component\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceReferencingComponentDTO - a model defined in Swagger

**active\_thread\_count**

Gets the active\_thread\_count of this ControllerServiceReferencingComponentDTO. The number of active threads for the referencing component.

**Returns** The active\_thread\_count of this ControllerServiceReferencingComponentDTO.

**Return type** int

```
attribute_map = {'active_thread_count': 'activeThreadCount', 'descriptors': 'descrip
```

**descriptors**

Gets the descriptors of this `ControllerServiceReferencingComponentDTO`. The descriptors for the component properties.

**Returns** The descriptors of this `ControllerServiceReferencingComponentDTO`.

**Return type** dict(str, *PropertyDescriptorDTO*)

**group\_id**

Gets the `group_id` of this `ControllerServiceReferencingComponentDTO`. The group id for the component referencing a controller service. If this component is another controller service or a reporting task, this field is blank.

**Returns** The `group_id` of this `ControllerServiceReferencingComponentDTO`.

**Return type** str

**id**

Gets the id of this `ControllerServiceReferencingComponentDTO`. The id of the component referencing a controller service.

**Returns** The id of this `ControllerServiceReferencingComponentDTO`.

**Return type** str

**name**

Gets the name of this `ControllerServiceReferencingComponentDTO`. The name of the component referencing a controller service.

**Returns** The name of this `ControllerServiceReferencingComponentDTO`.

**Return type** str

**properties**

Gets the properties of this `ControllerServiceReferencingComponentDTO`. The properties for the component.

**Returns** The properties of this `ControllerServiceReferencingComponentDTO`.

**Return type** dict(str, str)

**reference\_cycle**

Gets the `reference_cycle` of this `ControllerServiceReferencingComponentDTO`. If the referencing component represents a controller service, this indicates whether it has already been represented in this hierarchy.

**Returns** The `reference_cycle` of this `ControllerServiceReferencingComponentDTO`.

**Return type** bool

**reference\_type**

Gets the `reference_type` of this `ControllerServiceReferencingComponentDTO`. The type of reference this is.

**Returns** The `reference_type` of this `ControllerServiceReferencingComponentDTO`.

**Return type** str

**referencing\_components**

Gets the `referencing_components` of this `ControllerServiceReferencingComponentDTO`. If the referencing component represents a controller service, these are the components that reference it.

**Returns** The `referencing_components` of this `ControllerServiceReferencingComponentDTO`.

**Return type** list[*ControllerServiceReferencingComponentEntity*]

**state**

Gets the state of this ControllerServiceReferencingComponentDTO. The scheduled state of a processor or reporting task referencing a controller service. If this component is another controller service, this field represents the controller service state.

**Returns** The state of this ControllerServiceReferencingComponentDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'descriptors': 'dict(str, PropertyDes

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**type**

Gets the type of this ControllerServiceReferencingComponentDTO. The type of the component referencing a controller service in simple Java class name format without package name.

**Returns** The type of this ControllerServiceReferencingComponentDTO.

**Return type** str

**validation\_errors**

Gets the validation\_errors of this ControllerServiceReferencingComponentDTO. The validation errors for the component.

**Returns** The validation\_errors of this ControllerServiceReferencingComponentDTO.

**Return type** list[str]

**nipyapi.nifi.models.controller\_service\_referencing\_component\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceReferencingComponentEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_node_acknowledged': 'disconnected_node_acknowledged', 'id': 'id', 'operate_permissions': 'operate_permissions'}
```

**bulletins**

Gets the bulletins of this ControllerServiceReferencingComponentEntity. The bulletins for this component.

**Returns** The bulletins of this ControllerServiceReferencingComponentEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this ControllerServiceReferencingComponentEntity.

**Returns** The component of this ControllerServiceReferencingComponentEntity.

**Return type** *ControllerServiceReferencingComponentDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ControllerServiceReferencingComponentEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ControllerServiceReferencingComponentEntity.

**Return type** bool

**id**

Gets the id of this ControllerServiceReferencingComponentEntity. The id of the component.

**Returns** The id of this ControllerServiceReferencingComponentEntity.

**Return type** str

**operate\_permissions**

Gets the operate\_permissions of this ControllerServiceReferencingComponentEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this ControllerServiceReferencingComponentEntity.

**Return type** *PermissionsDTO*

#### **permissions**

Gets the permissions of this ControllerServiceReferencingComponentEntity. The permissions for this component.

**Returns** The permissions of this ControllerServiceReferencingComponentEntity.

**Return type** *PermissionsDTO*

#### **position**

Gets the position of this ControllerServiceReferencingComponentEntity. The position of this component in the UI if applicable.

**Returns** The position of this ControllerServiceReferencingComponentEntity.

**Return type** *PositionDTO*

#### **revision**

Gets the revision of this ControllerServiceReferencingComponentEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ControllerServiceReferencingComponentEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'ControllerServiceReferencingComponentEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

#### **uri**

Gets the uri of this ControllerServiceReferencingComponentEntity. The URI for futures requests to the component.

**Returns** The uri of this ControllerServiceReferencingComponentEntity.

**Return type** str

### **nipyapi.nifi.models.controller\_service\_referencing\_components\_entity module**

#### **NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_service\_referencing\_components\_entity.ControllerServiceReferencingComponentsEntity  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceReferencingComponentsEntity - a model defined in Swagger

**attribute\_map** = {'controller\_service\_referencing\_components': 'controllerServiceReferencingComponentsEntity'}

**controller\_service\_referencing\_components**

Gets the controller\_service\_referencing\_components of this ControllerServiceReferencingComponentsEntity.

**Returns** The controller\_service\_referencing\_components of this ControllerServiceReferencingComponentsEntity.

**Return type** list[*ControllerServiceReferencingComponentEntity*]

**swagger\_types** = {'controller\_service\_referencing\_components': 'list[ControllerServiceReferencingComponentEntity]'}  
**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.controller\_service\_types\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.controller\_service\_types\_entity.**ControllerServiceTypesEntity** (control\_service\_types\_entity)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceTypesEntity - a model defined in Swagger

**attribute\_map** = {'controller\_service\_types': 'controllerServiceTypes'}

**controller\_service\_types**

Gets the controller\_service\_types of this ControllerServiceTypesEntity.

**Returns** The controller\_service\_types of this ControllerServiceTypesEntity.

**Return type** list[*DocumentedTypeDTO*]

**swagger\_types** = {'controller\_service\_types': 'list[DocumentedTypeDTO]'}  
**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.controller\_services\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity(current_time=None, controller_services=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServicesEntity - a model defined in Swagger

```
attribute_map = {'controller_services': 'controllerServices', 'current_time': 'currentTime'}
```

```
controller_services
```

Gets the controller\_services of this ControllerServicesEntity.

**Returns** The controller\_services of this ControllerServicesEntity.

**Return type** list[*ControllerServiceEntity*]

```
current_time
```

Gets the current\_time of this ControllerServicesEntity. The current time on the system.

**Returns** The current\_time of this ControllerServicesEntity.

**Return type** str

```
swagger_types = {'controller_services': 'list[ControllerServiceEntity]', 'current_time': 'str'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

### nipyapi.nifi.models.controller\_status\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO(active_thread_count=None,  
terminated_thread_count=None,  
queued=None,  
flow_files_queued=None,  
bytes_queued=None,  
running_count=None,  
stopped_count=None,  
invalid_count=None,  
disabled_count=None,  
active_remote_port_count=None,  
inactive_remote_port_count=None,  
up_to_date_count=None,  
logically_modified_count=None,  
stale_count=None,  
logically_modified_and_stale_count=None,  
sync_failure_count=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerStatusDTO - a model defined in Swagger

**active\_remote\_port\_count**

Gets the active\_remote\_port\_count of this ControllerStatusDTO. The number of active remote ports in the NiFi.

**Returns** The active\_remote\_port\_count of this ControllerStatusDTO.

**Return type** int

**active\_thread\_count**

Gets the active\_thread\_count of this ControllerStatusDTO. The number of active threads in the NiFi.

**Returns** The active\_thread\_count of this ControllerStatusDTO.

**Return type** int

**attribute\_map** = {'active\_remote\_port\_count': 'activeRemotePortCount', 'active\_thread\_count': 'activeThreadCount'}

**bytes\_queued**

Gets the bytes\_queued of this ControllerStatusDTO. The size of the FlowFiles queued across the entire flow

**Returns** The bytes\_queued of this ControllerStatusDTO.

**Return type** int

**disabled\_count**

Gets the disabled\_count of this ControllerStatusDTO. The number of disabled components in the NiFi.

**Returns** The disabled\_count of this ControllerStatusDTO.

**Return type** int



**flow\_files\_queued**

Gets the flow\_files\_queued of this ControllerStatusDTO. The number of FlowFiles queued across the entire flow

**Returns** The flow\_files\_queued of this ControllerStatusDTO.

**Return type** int

**inactive\_remote\_port\_count**

Gets the inactive\_remote\_port\_count of this ControllerStatusDTO. The number of inactive remote ports in the NiFi.

**Returns** The inactive\_remote\_port\_count of this ControllerStatusDTO.

**Return type** int

**invalid\_count**

Gets the invalid\_count of this ControllerStatusDTO. The number of invalid components in the NiFi.

**Returns** The invalid\_count of this ControllerStatusDTO.

**Return type** int

**locally\_modified\_and\_stale\_count**

Gets the locally\_modified\_and\_stale\_count of this ControllerStatusDTO. The number of locally modified and stale versioned process groups in the NiFi.

**Returns** The locally\_modified\_and\_stale\_count of this ControllerStatusDTO.

**Return type** int

**locally\_modified\_count**

Gets the locally\_modified\_count of this ControllerStatusDTO. The number of locally modified versioned process groups in the NiFi.

**Returns** The locally\_modified\_count of this ControllerStatusDTO.

**Return type** int

**queued**

Gets the queued of this ControllerStatusDTO. The number of flowfiles queued in the NiFi.

**Returns** The queued of this ControllerStatusDTO.

**Return type** str

**running\_count**

Gets the running\_count of this ControllerStatusDTO. The number of running components in the NiFi.

**Returns** The running\_count of this ControllerStatusDTO.

**Return type** int

**stale\_count**

Gets the stale\_count of this ControllerStatusDTO. The number of stale versioned process groups in the NiFi.

**Returns** The stale\_count of this ControllerStatusDTO.

**Return type** int

**stopped\_count**

Gets the stopped\_count of this ControllerStatusDTO. The number of stopped components in the NiFi.

**Returns** The stopped\_count of this ControllerStatusDTO.

**Return type** int

```
swagger_types = {'active_remote_port_count': 'int', 'active_thread_count': 'int', 'b'
```

```
sync_failure_count
```

Gets the sync\_failure\_count of this ControllerStatusDTO. The number of versioned process groups in the NiFi that are unable to sync to a registry.

**Returns** The sync\_failure\_count of this ControllerStatusDTO.

**Return type** int

```
terminated_thread_count
```

Gets the terminated\_thread\_count of this ControllerStatusDTO. The number of terminated threads in the NiFi.

**Returns** The terminated\_thread\_count of this ControllerStatusDTO.

**Return type** int

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

```
up_to_date_count
```

Gets the up\_to\_date\_count of this ControllerStatusDTO. The number of up to date versioned process groups in the NiFi.

**Returns** The up\_to\_date\_count of this ControllerStatusDTO.

**Return type** int

## nipyapi.nifi.models.controller\_status\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity(controller_status=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerStatusEntity - a model defined in Swagger

```
attribute_map = {'controller_status': 'controllerStatus'}
```

```
controller_status
```

Gets the controller\_status of this ControllerStatusEntity.

**Returns** The controller\_status of this ControllerStatusEntity.

**Return type** *ControllerStatusDTO*

```
swagger_types = {'controller_status': 'ControllerStatusDTO'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.copy\_snippet\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.copy_snippet_request_entity.CopySnippetRequestEntity (snippet_id=None,
                                         origin_x=None,
                                         origin_y=None,
                                         disconnected_node_acknowledged=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CopySnippetRequestEntity - a model defined in Swagger

```
attribute_map = {'disconnected_node_acknowledged': 'disconnectedNodeAcknowledged', 'origin_x': 'originX', 'origin_y': 'originY', 'snippet_id': 'snippetId'}
```

#### disconnected\_node\_acknowledged

Gets the disconnected\_node\_acknowledged of this CopySnippetRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this CopySnippetRequestEntity.

**Return type** bool

#### origin\_x

Gets the origin\_x of this CopySnippetRequestEntity. The x coordinate of the origin of the bounding box where the new components will be placed.

**Returns** The origin\_x of this CopySnippetRequestEntity.

**Return type** float

#### origin\_y

Gets the origin\_y of this CopySnippetRequestEntity. The y coordinate of the origin of the bounding box where the new components will be placed.

**Returns** The origin\_y of this CopySnippetRequestEntity.

**Return type** float

#### snippet\_id

Gets the snippet\_id of this CopySnippetRequestEntity. The identifier of the snippet.

**Returns** The snippet\_id of this CopySnippetRequestEntity.

**Return type** str

```
swagger_types = {'disconnected_node_acknowledged': 'bool', 'origin_x': 'float', 'origin_y': 'float', 'snippet_id': 'str'}
```

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

## nipyapi.nifi.models.counter\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.counter_dto.CounterDTO (id=None, context=None,  
                                                name=None, value_count=None,  
                                                value=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CounterDTO - a model defined in Swagger

```
attribute_map = {'context': 'context', 'id': 'id', 'name': 'name', 'value': 'value'}
```

#### context

Gets the context of this CounterDTO. The context of the counter.

**Returns** The context of this CounterDTO.

**Return type** str

#### id

Gets the id of this CounterDTO. The id of the counter.

**Returns** The id of this CounterDTO.

**Return type** str

#### name

Gets the name of this CounterDTO. The name of the counter.

**Returns** The name of this CounterDTO.

**Return type** str

```
swagger_types = {'context': 'str', 'id': 'str', 'name': 'str', 'value': 'str', 'value_count': 'str'}
```

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

**value**  
Gets the value of this CounterDTO. The value of the counter.

**Returns** The value of this CounterDTO.

**Return type** str

**value\_count**

Gets the value\_count of this CounterDTO. The value count.

**Returns** The value\_count of this CounterDTO.

**Return type** int

**nipyapi.nifi.models.counter\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.counter\_entity.**CounterEntity**(counter=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CounterEntity - a model defined in Swagger

**attribute\_map** = {'counter': 'counter'}

**counter**

Gets the counter of this CounterEntity.

**Returns** The counter of this CounterEntity.

**Return type** *CounterDTO*

**swagger\_types** = {'counter': 'CounterDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.counters\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.counters\_dto.**CountersDTO**(aggregate\_snapshot=None, node\_snapshots=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CountersDTO - a model defined in Swagger

**aggregate\_snapshot**

Gets the aggregate\_snapshot of this CountersDTO. A Counters snapshot that represents the aggregate values of all nodes in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this represents the stats of the single instance.

**Returns** The aggregate\_snapshot of this CountersDTO.

**Return type** *CountersSnapshotDTO*

```
attribute_map = {'aggregate_snapshot': 'aggregateSnapshot', 'node_snapshots': 'nodes'
```

**node\_snapshots**

Gets the node\_snapshots of this CountersDTO. A Counters snapshot for each node in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this may be null.

**Returns** The node\_snapshots of this CountersDTO.

**Return type** list[*NodeCountersSnapshotDTO*]

```
swagger_types = {'aggregate_snapshot': 'CountersSnapshotDTO', 'node_snapshots': 'lis
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

**nipyapi.nifi.models.counters\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.counters_entity.CountersEntity(counters=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CountersEntity - a model defined in Swagger

```
attribute_map = {'counters': 'counters'}
```

**counters**

Gets the counters of this CountersEntity.

**Returns** The counters of this CountersEntity.

**Return type** *CountersDTO*

```
swagger_types = {'counters': 'CountersDTO'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.counters\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.counters_snapshot_dto.CountersSnapshotDTO(generated=None,
                                                                    coun-
                                                                    ters=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CountersSnapshotDTO - a model defined in Swagger

```
attribute_map = {'counters':  'counters', 'generated':  'generated'}
```

#### counters

Gets the counters of this CountersSnapshotDTO. All counters in the NiFi.

**Returns** The counters of this CountersSnapshotDTO.

**Return type** list[*CounterDTO*]

#### generated

Gets the generated of this CountersSnapshotDTO. The timestamp when the report was generated.

**Returns** The generated of this CountersSnapshotDTO.

**Return type** str

```
swagger_types = {'counters':  'list[CounterDTO]', 'generated':  'str'}
```

#### to\_dict()

Returns the model properties as a dict

#### to\_str()

Returns the string representation of the model

## nipyapi.nifi.models.create\_active\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.create_active_request_entity.CreateActiveRequestEntity(process_group,
                                                                                   dis-
                                                                                   con-
                                                                                   nected_node_
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CreateActiveRequestEntity - a model defined in Swagger

```
attribute_map = {'disconnected_node_acknowledged': 'disconnectedNodeAcknowledged', 'p
```

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this CreateActiveRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this CreateActiveRequestEntity.

**Return type** bool

**process\_group\_id**

Gets the process\_group\_id of this CreateActiveRequestEntity. The Process Group ID that this active request will update

**Returns** The process\_group\_id of this CreateActiveRequestEntity.

**Return type** str

```
swagger_types = {'disconnected_node_acknowledged': 'bool', 'process_group_id': 'str'
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.create\_template\_request\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity(name=N
de-
scrip-
tion=No
snip-
pet_id=
dis-
con-
nected_
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CreateTemplateRequestEntity - a model defined in Swagger

```
attribute_map = {'description': 'description', 'disconnected_node_acknowledged': 'di
```

**description**

Gets the description of this CreateTemplateRequestEntity. The description of the template.

**Returns** The description of this CreateTemplateRequestEntity.

**Return type** str



**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this CreateTemplateRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this CreateTemplateRequestEntity.

**Return type** bool

**name**

Gets the name of this CreateTemplateRequestEntity. The name of the template.

**Returns** The name of this CreateTemplateRequestEntity.

**Return type** str

**snippet\_id**

Gets the snippet\_id of this CreateTemplateRequestEntity. The identifier of the snippet.

**Returns** The snippet\_id of this CreateTemplateRequestEntity.

**Return type** str

**swagger\_types** = {'description': 'str', 'disconnected\_node\_acknowledged': 'bool', 'name': 'str', 'snippet\_id': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.current\_user\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.current_user_entity.CurrentUserEntity (identity=None,
                                                                    anony-
                                                                    mous=None,
                                                                    prove-
                                                                    nance_permissions=None,
                                                                    coun-
                                                                    ters_permissions=None,
                                                                    ten-
                                                                    ants_permissions=None,
                                                                    con-
                                                                    troller_permissions=None,
                                                                    poli-
                                                                    cies_permissions=None,
                                                                    sys-
                                                                    tem_permissions=None,
                                                                    parame-
                                                                    ter_context_permissions=None,
                                                                    re-
                                                                    stricted_components_permissions=None,
                                                                    compo-
                                                                    nent_restriction_permissions=None,
                                                                    can_version_flows=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CurrentUserEntity - a model defined in Swagger

#### **anonymous**

Gets the anonymous of this CurrentUserEntity. Whether the current user is anonymous.

**Returns** The anonymous of this CurrentUserEntity.

**Return type** bool

```
attribute_map = {'anonymous': 'anonymous', 'can_version_flows': 'canVersionFlows', 'can_version_flows': 'canVersionFlows'}
```

#### **can\_version\_flows**

Gets the can\_version\_flows of this CurrentUserEntity. Whether the current user can version flows.

**Returns** The can\_version\_flows of this CurrentUserEntity.

**Return type** bool

#### **component\_restriction\_permissions**

Gets the component\_restriction\_permissions of this CurrentUserEntity. Permissions for specific component restrictions.

**Returns** The component\_restriction\_permissions of this CurrentUserEntity.

**Return type** list[ComponentRestrictionPermissionDTO]

#### **controller\_permissions**

Gets the controller\_permissions of this CurrentUserEntity. Permissions for accessing the controller.

**Returns** The controller\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **counters\_permissions**

Gets the counters\_permissions of this CurrentUserEntity. Permissions for accessing counters.

**Returns** The counters\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **identity**

Gets the identity of this CurrentUserEntity. The user identity being serialized.

**Returns** The identity of this CurrentUserEntity.

**Return type** str

#### **parameter\_context\_permissions**

Gets the parameter\_context\_permissions of this CurrentUserEntity. Permissions for accessing parameter contexts.

**Returns** The parameter\_context\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **policies\_permissions**

Gets the policies\_permissions of this CurrentUserEntity. Permissions for accessing the policies.

**Returns** The policies\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **provenance\_permissions**

Gets the provenance\_permissions of this CurrentUserEntity. Permissions for querying provenance.

**Returns** The provenance\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **restricted\_components\_permissions**

Gets the restricted\_components\_permissions of this CurrentUserEntity. Permissions for accessing restricted components. Note: the read permission are not used and will always be false.

**Returns** The restricted\_components\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

**swagger\_types** = {'anonymous': 'bool', 'can\_version\_flows': 'bool', 'component\_restri

#### **system\_permissions**

Gets the system\_permissions of this CurrentUserEntity. Permissions for accessing system.

**Returns** The system\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **tenants\_permissions**

Gets the tenants\_permissions of this CurrentUserEntity. Permissions for accessing tenants.

**Returns** The tenants\_permissions of this CurrentUserEntity.

**Return type** *PermissionsDTO*

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## **nipyapi.nifi.models.difference\_dto module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.difference_dto.DifferenceDTO (difference_type=None, difference=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

DifferenceDTO - a model defined in Swagger

```
attribute_map = {'difference': 'difference', 'difference_type': 'differenceType'}
```

**difference**

Gets the difference of this DifferenceDTO. Description of the difference

**Returns** The difference of this DifferenceDTO.

**Return type** str

**difference\_type**

Gets the difference\_type of this DifferenceDTO. The type of difference

**Returns** The difference\_type of this DifferenceDTO.

**Return type** str

```
swagger_types = {'difference': 'str', 'difference_type': 'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## **nipyapi.nifi.models.dimensions\_dto module**

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.dimensions_dto.DimensionsDTO (width=None, height=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

DimensionsDTO - a model defined in Swagger

```
attribute_map = {'height': 'height', 'width': 'width'}
```

**height**

Gets the height of this DimensionsDTO. The height of the label in pixels when at a 1:1 scale.

**Returns** The height of this DimensionsDTO.

**Return type** float

```
swagger_types = {'height': 'float', 'width': 'float'}
```

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

**width**  
Gets the width of this DimensionsDTO. The width of the label in pixels when at a 1:1 scale.

**Returns** The width of this DimensionsDTO.

**Return type** float

### nipyapi.nifi.models.documented\_type\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO (type=None,
                                                                    bundle=None,
                                                                    con-
                                                                    troller_service_apis=None,
                                                                    descrip-
                                                                    tion=None, re-
                                                                    stricted=None,
                                                                    us-
                                                                    age_restriction=None,
                                                                    ex-
                                                                    plicit_restrictions=None,
                                                                    depreca-
                                                                    tion_reason=None,
                                                                    tags=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

DocumentedTypeDTO - a model defined in Swagger

```
attribute_map = {'bundle': 'bundle', 'controller_service_apis': 'controllerServiceApis'}
```

#### bundle

Gets the bundle of this DocumentedTypeDTO. The details of the artifact that bundled this type.

**Returns** The bundle of this DocumentedTypeDTO.

**Return type** *BundleDTO*

#### controller\_service\_apis

Gets the controller\_service\_apis of this DocumentedTypeDTO. If this type represents a ControllerService, this lists the APIs it implements.

**Returns** The controller\_service\_apis of this DocumentedTypeDTO.

**Return type** list[*ControllerServiceApiDTO*]

**deprecation\_reason**

Gets the `deprecation_reason` of this `DocumentedTypeDTO`. The description of why the usage of this component is restricted.

**Returns** The `deprecation_reason` of this `DocumentedTypeDTO`.

**Return type** `str`

**description**

Gets the description of this `DocumentedTypeDTO`. The description of the type.

**Returns** The description of this `DocumentedTypeDTO`.

**Return type** `str`

**explicit\_restrictions**

Gets the `explicit_restrictions` of this `DocumentedTypeDTO`. An optional collection of explicit restrictions. If specified, these explicit restrictions will be enforced.

**Returns** The `explicit_restrictions` of this `DocumentedTypeDTO`.

**Return type** `list[ExplicitRestrictionDTO]`

**restricted**

Gets the `restricted` of this `DocumentedTypeDTO`. Whether this type is restricted.

**Returns** The `restricted` of this `DocumentedTypeDTO`.

**Return type** `bool`

**swagger\_types** = {'bundle': 'BundleDTO', 'controller\_service\_apis': 'list[ControllerS

**tags**

Gets the tags of this `DocumentedTypeDTO`. The tags associated with this type.

**Returns** The tags of this `DocumentedTypeDTO`.

**Return type** `list[str]`

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this `DocumentedTypeDTO`. The fully qualified name of the type.

**Returns** The type of this `DocumentedTypeDTO`.

**Return type** `str`

**usage\_restriction**

Gets the `usage_restriction` of this `DocumentedTypeDTO`. The optional description of why the usage of this component is restricted.

**Returns** The `usage_restriction` of this `DocumentedTypeDTO`.

**Return type** `str`

## nipyapi.nifi.models.drop\_request\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.drop_request_dto.DropRequestDTO (id=None,
                                                         uri=None,      submis-
                                                         sion_time=None,
                                                         last_updated=None,
                                                         per-
                                                         cent_completed=None,
                                                         finished=None,   fail-
                                                         ure_reason=None,
                                                         current_count=None,
                                                         current_size=None,
                                                         current=None,   orig-
                                                         inal_count=None,
                                                         original_size=None,
                                                         original=None,
                                                         dropped_count=None,
                                                         dropped_size=None,
                                                         dropped=None,
                                                         state=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

DropRequestDTO - a model defined in Swagger

```
attribute_map = {'current':  'current', 'current_count':  'currentCount', 'current_size':  'currentSize'}
```

**current**

Gets the current of this DropRequestDTO. The count and size of flow files currently queued.

**Returns** The current of this DropRequestDTO.

**Return type** str

**current\_count**

Gets the current\_count of this DropRequestDTO. The number of flow files currently queued.

**Returns** The current\_count of this DropRequestDTO.

**Return type** int

**current\_size**

Gets the current\_size of this DropRequestDTO. The size of flow files currently queued in bytes.

**Returns** The current\_size of this DropRequestDTO.

**Return type** int

**dropped**

Gets the dropped of this DropRequestDTO. The count and size of flow files that have been dropped thus far.

**Returns** The dropped of this DropRequestDTO.

**Return type** str

**dropped\_count**

Gets the dropped\_count of this DropRequestDTO. The number of flow files that have been dropped thus far.

**Returns** The dropped\_count of this DropRequestDTO.

**Return type** int

**dropped\_size**

Gets the dropped\_size of this DropRequestDTO. The size of flow files that have been dropped thus far in bytes.

**Returns** The dropped\_size of this DropRequestDTO.

**Return type** int

**failure\_reason**

Gets the failure\_reason of this DropRequestDTO. The reason, if any, that this drop request failed.

**Returns** The failure\_reason of this DropRequestDTO.

**Return type** str

**finished**

Gets the finished of this DropRequestDTO. Whether the query has finished.

**Returns** The finished of this DropRequestDTO.

**Return type** bool

**id**

Gets the id of this DropRequestDTO. The id for this drop request.

**Returns** The id of this DropRequestDTO.

**Return type** str

**last\_updated**

Gets the last\_updated of this DropRequestDTO. The last time this drop request was updated.

**Returns** The last\_updated of this DropRequestDTO.

**Return type** str

**original**

Gets the original of this DropRequestDTO. The count and size of flow files to be dropped as a result of this request.

**Returns** The original of this DropRequestDTO.

**Return type** str

**original\_count**

Gets the original\_count of this DropRequestDTO. The number of flow files to be dropped as a result of this request.

**Returns** The original\_count of this DropRequestDTO.

**Return type** int

**original\_size**

Gets the original\_size of this DropRequestDTO. The size of flow files to be dropped as a result of this request in bytes.

**Returns** The original\_size of this DropRequestDTO.

**Return type** int





**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

## nipyapi.nifi.models.flow\_breadcrumb\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO(id=None,
                                                                name=None,
                                                                ver-
                                                                sion_control_information=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowBreadcrumbDTO - a model defined in Swagger

```
attribute_map = {'id': 'id', 'name': 'name', 'version_control_information': 'version
```

**id**

Gets the id of this FlowBreadcrumbDTO. The id of the group.

**Returns** The id of this FlowBreadcrumbDTO.

**Return type** str

**name**

Gets the name of this FlowBreadcrumbDTO. The id of the group.

**Returns** The name of this FlowBreadcrumbDTO.

**Return type** str

```
swagger_types = {'id': 'str', 'name': 'str', 'version_control_information': 'Versio
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**version\_control\_information**

Gets the version\_control\_information of this FlowBreadcrumbDTO. The process group version control information or null if not version controlled.

**Returns** The version\_control\_information of this FlowBreadcrumbDTO.

**Return type** *VersionControlInformationDTO*

## nipyapi.nifi.models.flow\_breadcrumb\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity(id=None,
                                                                    per-
                                                                    mis-
                                                                    sions=None,
                                                                    ver-
                                                                    sioned_flow_state=None,
                                                                    bread-
                                                                    crumb=None,
                                                                    par-
                                                                    ent_breadcrumb=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowBreadcrumbEntity - a model defined in Swagger

```
attribute_map = {'breadcrumb': 'breadcrumb', 'id': 'id', 'parent_breadcrumb': 'parent_breadcrumb'}
```

#### breadcrumb

Gets the breadcrumb of this FlowBreadcrumbEntity. This breadcrumb.

**Returns** The breadcrumb of this FlowBreadcrumbEntity.

**Return type** *FlowBreadcrumbDTO*

#### id

Gets the id of this FlowBreadcrumbEntity. The id of this ancestor ProcessGroup.

**Returns** The id of this FlowBreadcrumbEntity.

**Return type** str

#### parent\_breadcrumb

Gets the parent\_breadcrumb of this FlowBreadcrumbEntity. The parent breadcrumb for this breadcrumb.

**Returns** The parent\_breadcrumb of this FlowBreadcrumbEntity.

**Return type** *FlowBreadcrumbEntity*

#### permissions

Gets the permissions of this FlowBreadcrumbEntity. The permissions for this ancestor ProcessGroup.

**Returns** The permissions of this FlowBreadcrumbEntity.

**Return type** *PermissionsDTO*

```
swagger_types = {'breadcrumb': 'FlowBreadcrumbDTO', 'id': 'str', 'parent_breadcrumb': 'FlowBreadcrumbEntity'}
```

#### to\_dict()

Returns the model properties as a dict

#### to\_str()

Returns the string representation of the model

**versioned\_flow\_state**

Gets the versioned\_flow\_state of this FlowBreadcrumbEntity. The current state of the Process Group, as it relates to the Versioned Flow

**Returns** The versioned\_flow\_state of this FlowBreadcrumbEntity.

**Return type** str

**nipyapi.nifi.models.flow\_comparison\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_comparison_entity.FlowComparisonEntity(component_differences=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowComparisonEntity - a model defined in Swagger

```
attribute_map = {'component_differences': 'componentDifferences'}
```

**component\_differences**

Gets the component\_differences of this FlowComparisonEntity. The list of differences for each component in the flow that is not the same between the two flows

**Returns** The component\_differences of this FlowComparisonEntity.

**Return type** list[*ComponentDifferenceDTO*]

```
swagger_types = {'component_differences': 'list[ComponentDifferenceDTO]'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

**nipyapi.nifi.models.flow\_configuration\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO (supports_managed_authorizer,
                                                                    supports_configurable_authorizer,
                                                                    supports_configurable_users_and_groups,
                                                                    auto_refresh_interval_seconds,
                                                                    current_time=None,
                                                                    time_offset=None,
                                                                    default_back_pressure_object_threshold=None,
                                                                    default_back_pressure_data_size_threshold=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowConfigurationDTO - a model defined in Swagger

```
attribute_map = {'auto_refresh_interval_seconds': 'autoRefreshIntervalSeconds', 'current_time': 'currentTime', 'time_offset': 'timeOffset', 'default_back_pressure_object_threshold': 'defaultBackPressureObjectThreshold', 'default_back_pressure_data_size_threshold': 'defaultBackPressureDataSizeThreshold'}
```

#### **auto\_refresh\_interval\_seconds**

Gets the auto\_refresh\_interval\_seconds of this FlowConfigurationDTO. The interval in seconds between the automatic NiFi refresh requests.

**Returns** The auto\_refresh\_interval\_seconds of this FlowConfigurationDTO.

**Return type** int

#### **current\_time**

Gets the current\_time of this FlowConfigurationDTO. The current time on the system.

**Returns** The current\_time of this FlowConfigurationDTO.

**Return type** str

#### **default\_back\_pressure\_data\_size\_threshold**

Gets the default\_back\_pressure\_data\_size\_threshold of this FlowConfigurationDTO. The default back pressure data size threshold.

**Returns** The default\_back\_pressure\_data\_size\_threshold of this FlowConfigurationDTO.

**Return type** str

#### **default\_back\_pressure\_object\_threshold**

Gets the default\_back\_pressure\_object\_threshold of this FlowConfigurationDTO. The default back pressure object threshold.

**Returns** The default\_back\_pressure\_object\_threshold of this FlowConfigurationDTO.

**Return type** int

#### **supports\_configurable\_authorizer**

Gets the supports\_configurable\_authorizer of this FlowConfigurationDTO. Whether this NiFi supports a configurable authorizer.

**Returns** The supports\_configurable\_authorizer of this FlowConfigurationDTO.

**Return type** bool

#### **supports\_configurable\_users\_and\_groups**

Gets the supports\_configurable\_users\_and\_groups of this FlowConfigurationDTO. Whether this NiFi supports configurable users and groups.

**Returns** The supports\_configurable\_users\_and\_groups of this FlowConfigurationDTO.

**Return type** bool

**supports\_managed\_authorizer**

Gets the supports\_managed\_authorizer of this FlowConfigurationDTO. Whether this NiFi supports a managed authorizer. Managed authorizers can visualize users, groups, and policies in the UI.

**Returns** The supports\_managed\_authorizer of this FlowConfigurationDTO.

**Return type** bool

**swagger\_types** = {'auto\_refresh\_interval\_seconds': 'int', 'current\_time': 'str', 'def

**time\_offset**

Gets the time\_offset of this FlowConfigurationDTO. The time offset of the system.

**Returns** The time\_offset of this FlowConfigurationDTO.

**Return type** int

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.flow\_configuration\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.flow\_configuration\_entity.**FlowConfigurationEntity**(flow\_configuration=*None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowConfigurationEntity - a model defined in Swagger

**attribute\_map** = {'flow\_configuration': 'flowConfiguration'}

**flow\_configuration**

Gets the flow\_configuration of this FlowConfigurationEntity. The controller configuration.

**Returns** The flow\_configuration of this FlowConfigurationEntity.

**Return type** *FlowConfigurationDTO*

**swagger\_types** = {'flow\_configuration': 'FlowConfigurationDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.flow\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_dto.FlowDTO(process_groups=None, remote_process_groups=None, processors=None, input_ports=None, output_ports=None, connections=None, labels=None, funnels=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowDTO - a model defined in Swagger

```
attribute_map = {'connections': 'connections', 'funnels': 'funnels', 'input_ports':
```

#### **connections**

Gets the connections of this FlowDTO. The connections in this flow.

**Returns** The connections of this FlowDTO.

**Return type** list[*ConnectionEntity*]

#### **funnels**

Gets the funnels of this FlowDTO. The funnels in this flow.

**Returns** The funnels of this FlowDTO.

**Return type** list[*FunnelEntity*]

#### **input\_ports**

Gets the input\_ports of this FlowDTO. The input ports in this flow.

**Returns** The input\_ports of this FlowDTO.

**Return type** list[*PortEntity*]

#### **labels**

Gets the labels of this FlowDTO. The labels in this flow.

**Returns** The labels of this FlowDTO.

**Return type** list[*LabelEntity*]

#### **output\_ports**

Gets the output\_ports of this FlowDTO. The output ports in this flow.

**Returns** The output\_ports of this FlowDTO.

**Return type** list[*PortEntity*]

#### **process\_groups**

Gets the process\_groups of this FlowDTO. The process groups in this flow.

**Returns** The process\_groups of this FlowDTO.

**Return type** list[*ProcessGroupEntity*]

**processors**

Gets the processors of this FlowDTO. The processors in this flow.

**Returns** The processors of this FlowDTO.

**Return type** list[*ProcessorEntity*]

**remote\_process\_groups**

Gets the remote\_process\_groups of this FlowDTO. The remote process groups in this flow.

**Returns** The remote\_process\_groups of this FlowDTO.

**Return type** list[*RemoteProcessGroupEntity*]

**swagger\_types** = {'connections': 'list[ConnectionEntity]', 'funnels': 'list[FunnelEnt

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.flow\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.flow\_entity.**FlowEntity** (*flow=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowEntity - a model defined in Swagger

**attribute\_map** = {'flow': 'flow'}

**flow**

Gets the flow of this FlowEntity.

**Returns** The flow of this FlowEntity.

**Return type** *FlowDTO*

**swagger\_types** = {'flow': 'FlowDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.flow\_file\_dto module**

## NiFi Rest API



The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_file_dto.FlowFileDTO (uri=None, uuid=None,
                                                    filename=None, position=None,
                                                    size=None,
                                                    queued_duration=None,
                                                    lineage_duration=None,
                                                    penalty_expires_in=None,
                                                    cluster_node_id=None, cluster_node_address=None,
                                                    attributes=None, content_claim_section=None,
                                                    content_claim_container=None,
                                                    content_claim_identifier=None,
                                                    content_claim_offset=None,
                                                    content_claim_file_size=None,
                                                    content_claim_file_size_bytes=None,
                                                    penalized=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowFileDTO - a model defined in Swagger

```
attribute_map = {'attributes': 'attributes', 'cluster_node_address': 'clusterNodeAddress'}
```

**attributes**

Gets the attributes of this FlowFileDTO. The FlowFile attributes.

**Returns** The attributes of this FlowFileDTO.

**Return type** dict(str, str)

**cluster\_node\_address**

Gets the cluster\_node\_address of this FlowFileDTO. The label for the node where this FlowFile resides.

**Returns** The cluster\_node\_address of this FlowFileDTO.

**Return type** str

**cluster\_node\_id**

Gets the cluster\_node\_id of this FlowFileDTO. The id of the node where this FlowFile resides.

**Returns** The cluster\_node\_id of this FlowFileDTO.

**Return type** str

**content\_claim\_container**

Gets the content\_claim\_container of this FlowFileDTO. The container in which the content claim lives.

**Returns** The content\_claim\_container of this FlowFileDTO.

**Return type** str

**content\_claim\_file\_size**

Gets the content\_claim\_file\_size of this FlowFileDTO. The file size of the content claim formatted.

**Returns** The content\_claim\_file\_size of this FlowFileDTO.

**Return type** str

**content\_claim\_file\_size\_bytes**

Gets the content\_claim\_file\_size\_bytes of this FlowFileDTO. The file size of the content claim in bytes.

**Returns** The content\_claim\_file\_size\_bytes of this FlowFileDTO.

**Return type** int

**content\_claim\_identifier**

Gets the content\_claim\_identifier of this FlowFileDTO. The identifier of the content claim.

**Returns** The content\_claim\_identifier of this FlowFileDTO.

**Return type** str

**content\_claim\_offset**

Gets the content\_claim\_offset of this FlowFileDTO. The offset into the content claim where the flowfile's content begins.

**Returns** The content\_claim\_offset of this FlowFileDTO.

**Return type** int

**content\_claim\_section**

Gets the content\_claim\_section of this FlowFileDTO. The section in which the content claim lives.

**Returns** The content\_claim\_section of this FlowFileDTO.

**Return type** str

**filename**

Gets the filename of this FlowFileDTO. The FlowFile filename.

**Returns** The filename of this FlowFileDTO.

**Return type** str

**lineage\_duration**

Gets the lineage\_duration of this FlowFileDTO. Duration since the FlowFile's greatest ancestor entered the flow.

**Returns** The lineage\_duration of this FlowFileDTO.

**Return type** int

**penalized**

Gets the penalized of this FlowFileDTO. If the FlowFile is penalized.

**Returns** The penalized of this FlowFileDTO.

**Return type** bool

**penalty\_expires\_in**

Gets the penalty\_expires\_in of this FlowFileDTO. How long in milliseconds until the FlowFile penalty expires.

**Returns** The penalty\_expires\_in of this FlowFileDTO.

**Return type** int

**position**

Gets the position of this FlowFileDTO. The FlowFile's position in the queue.

**Returns** The position of this FlowFileDTO.

**Return type** int

**queued\_duration**

Gets the queued\_duration of this FlowFileDTO. How long this FlowFile has been enqueued.

**Returns** The queued\_duration of this FlowFileDTO.

**Return type** int

**size**

Gets the size of this FlowFileDTO. The FlowFile file size.

**Returns** The size of this FlowFileDTO.

**Return type** int

**swagger\_types** = {'attributes': 'dict(str, str)', 'cluster\_node\_address': 'str', 'clu

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this FlowFileDTO. The URI that can be used to access this FlowFile.

**Returns** The uri of this FlowFileDTO.

**Return type** str

**uuid**

Gets the uuid of this FlowFileDTO. The FlowFile UUID.

**Returns** The uuid of this FlowFileDTO.

**Return type** str

**nipyapi.nifi.models.flow\_file\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.flow\_file\_entity.**FlowFileEntity** (*flow\_file=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowFileEntity - a model defined in Swagger

**attribute\_map** = {'flow\_file': 'flowFile'}

**flow\_file**

Gets the flow\_file of this FlowFileEntity.

**Returns** The flow\_file of this FlowFileEntity.

**Return type** *FlowFileDTO*

**swagger\_types** = {'flow\_file': 'FlowFileDTO'}

```
to_dict()  
    Returns the model properties as a dict  
  
to_str()  
    Returns the string representation of the model
```

## nipyapi.nifi.models.flow\_file\_summary\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_file_summary_dto.FlowFileSummaryDTO(uri=None,  
                                                                    uuid=None,  
                                                                    file-  
                                                                    name=None,  
                                                                    posi-  
                                                                    tion=None,  
                                                                    size=None,  
                                                                    queued_duration=None,  
                                                                    lin-  
                                                                    eage_duration=None,  
                                                                    penalty_expires_in=None,  
                                                                    clus-  
                                                                    ter_node_id=None,  
                                                                    clus-  
                                                                    ter_node_address=None,  
                                                                    penal-  
                                                                    ized=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowFileSummaryDTO - a model defined in Swagger

```
attribute_map = {'cluster_node_address': 'clusterNodeAddress', 'cluster_node_id': 'c
```

#### **cluster\_node\_address**

Gets the cluster\_node\_address of this FlowFileSummaryDTO. The label for the node where this FlowFile resides.

**Returns** The cluster\_node\_address of this FlowFileSummaryDTO.

**Return type** str

#### **cluster\_node\_id**

Gets the cluster\_node\_id of this FlowFileSummaryDTO. The id of the node where this FlowFile resides.

**Returns** The cluster\_node\_id of this FlowFileSummaryDTO.

**Return type** str

#### **filename**

Gets the filename of this FlowFileSummaryDTO. The FlowFile filename.

**Returns** The filename of this FlowFileSummaryDTO.

**Return type** str

#### **lineage\_duration**

Gets the lineage\_duration of this FlowFileSummaryDTO. Duration since the FlowFile's greatest ancestor entered the flow.

**Returns** The lineage\_duration of this FlowFileSummaryDTO.

**Return type** int

#### **penalized**

Gets the penalized of this FlowFileSummaryDTO. If the FlowFile is penalized.

**Returns** The penalized of this FlowFileSummaryDTO.

**Return type** bool

#### **penalty\_expires\_in**

Gets the penalty\_expires\_in of this FlowFileSummaryDTO. How long in milliseconds until the FlowFile penalty expires.

**Returns** The penalty\_expires\_in of this FlowFileSummaryDTO.

**Return type** int

#### **position**

Gets the position of this FlowFileSummaryDTO. The FlowFile's position in the queue.

**Returns** The position of this FlowFileSummaryDTO.

**Return type** int

#### **queued\_duration**

Gets the queued\_duration of this FlowFileSummaryDTO. How long this FlowFile has been enqueued.

**Returns** The queued\_duration of this FlowFileSummaryDTO.

**Return type** int

#### **size**

Gets the size of this FlowFileSummaryDTO. The FlowFile file size.

**Returns** The size of this FlowFileSummaryDTO.

**Return type** int

**swagger\_types** = {'cluster\_node\_address': 'str', 'cluster\_node\_id': 'str', 'filename'

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **uri**

Gets the uri of this FlowFileSummaryDTO. The URI that can be used to access this FlowFile.

**Returns** The uri of this FlowFileSummaryDTO.

**Return type** str

#### **uuid**

Gets the uuid of this FlowFileSummaryDTO. The FlowFile UUID.

**Returns** The uuid of this FlowFileSummaryDTO.

**Return type** str

## nipyapi.nifi.models.flow\_snippet\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO (process_groups=None,
                                                         re-
                                                         mote_process_groups=None,
                                                         processors=None, in-
                                                         put_ports=None,
                                                         output_ports=None,
                                                         connections=None,
                                                         labels=None,      fun-
                                                         nels=None,      con-
                                                         troller_services=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FlowSnippetDTO - a model defined in Swagger

```
attribute_map = {'connections':  'connections', 'controller_services':  'controllerSer
```

#### **connections**

Gets the connections of this FlowSnippetDTO. The connections in this flow snippet.

**Returns** The connections of this FlowSnippetDTO.

**Return type** list[[ConnectionDTO](#)]

#### **controller\_services**

Gets the controller\_services of this FlowSnippetDTO. The controller services in this flow snippet.

**Returns** The controller\_services of this FlowSnippetDTO.

**Return type** list[[ControllerServiceDTO](#)]

#### **funnels**

Gets the funnels of this FlowSnippetDTO. The funnels in this flow snippet.

**Returns** The funnels of this FlowSnippetDTO.

**Return type** list[[FunnelDTO](#)]

#### **input\_ports**

Gets the input\_ports of this FlowSnippetDTO. The input ports in this flow snippet.

**Returns** The input\_ports of this FlowSnippetDTO.

**Return type** list[[PortDTO](#)]

#### **labels**

Gets the labels of this FlowSnippetDTO. The labels in this flow snippet.

**Returns** The labels of this FlowSnippetDTO.

**Return type** list[[LabelDTO](#)]

**output\_ports**

Gets the output\_ports of this FlowSnippetDTO. The output ports in this flow snippet.

**Returns** The output\_ports of this FlowSnippetDTO.

**Return type** list[PortDTO]

**process\_groups**

Gets the process\_groups of this FlowSnippetDTO. The process groups in this flow snippet.

**Returns** The process\_groups of this FlowSnippetDTO.

**Return type** list[ProcessGroupDTO]

**processors**

Gets the processors of this FlowSnippetDTO. The processors in this flow snippet.

**Returns** The processors of this FlowSnippetDTO.

**Return type** list[ProcessorDTO]

**remote\_process\_groups**

Gets the remote\_process\_groups of this FlowSnippetDTO. The remote process groups in this flow snippet.

**Returns** The remote\_process\_groups of this FlowSnippetDTO.

**Return type** list[RemoteProcessGroupDTO]

**swagger\_types** = {'connections': 'list[ConnectionDTO]', 'controller\_services': 'list[ControllerServiceDTO]'

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.funnel\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.funnel_dto.FunnelDTO(id=None, versioned_component_id=None, parent_group_id=None, position=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FunnelDTO - a model defined in Swagger

**attribute\_map** = {'id': 'id', 'parent\_group\_id': 'parentGroupId', 'position': 'position'}

**id**

Gets the id of this FunnelDTO. The id of the component.

**Returns** The id of this FunnelDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this FunnelDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this FunnelDTO.

**Return type** str

**position**

Gets the position of this FunnelDTO. The position of this component in the UI if applicable.

**Returns** The position of this FunnelDTO.

**Return type** *PositionDTO*

```
swagger_types = {'id': 'str', 'parent_group_id': 'str', 'position': 'PositionDTO',
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this FunnelDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this FunnelDTO.

**Return type** str

## nipyapi.nifi.models.funnel\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.funnel_entity.FunnelEntity(revision=None, id=None,
                                                    uri=None, position=None,
                                                    permissions=None, bul-
                                                    letins=None, discon-
                                                    nected_node_acknowledged=None,
                                                    component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FunnelEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this FunnelEntity. The bulletins for this component.

**Returns** The bulletins of this FunnelEntity.

**Return type** list[*BulletinEntity*]



**component**

Gets the component of this FunnelEntity.

**Returns** The component of this FunnelEntity.

**Return type** *FunnelDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this FunnelEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this FunnelEntity.

**Return type** bool

**id**

Gets the id of this FunnelEntity. The id of the component.

**Returns** The id of this FunnelEntity.

**Return type** str

**permissions**

Gets the permissions of this FunnelEntity. The permissions for this component.

**Returns** The permissions of this FunnelEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this FunnelEntity. The position of this component in the UI if applicable.

**Returns** The position of this FunnelEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this FunnelEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this FunnelEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'FunnelDTO', 'disconnected\_node\_acknowledged': 'bool', 'id': 'str', 'permissions': 'PermissionsDTO', 'position': 'PositionDTO', 'revision': 'RevisionDTO', 'uri': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this FunnelEntity. The URI for futures requests to the component.

**Returns** The uri of this FunnelEntity.

**Return type** str

**nipyapi.nifi.models.funnels\_entity module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.funnels_entity.FunnelsEntity (funnels=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

FunnelsEntity - a model defined in Swagger

```
attribute_map = {'funnels': 'funnels'}
```

```
funnels
```

Gets the funnels of this FunnelsEntity.

**Returns** The funnels of this FunnelsEntity.

**Return type** list[*FunnelEntity*]

```
swagger_types = {'funnels': 'list[FunnelEntity]'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

### nipyapi.nifi.models.garbage\_collection\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO (name=None,
                                                                    col-
                                                                    lec-
                                                                    tion_count=None,
                                                                    col-
                                                                    lec-
                                                                    tion_time=None,
                                                                    col-
                                                                    lec-
                                                                    tion_millis=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

GarbageCollectionDTO - a model defined in Swagger

```
attribute_map = {'collection_count': 'collectionCount', 'collection_millis': 'collectionMillis'}
```

```
collection_count
```

Gets the collection\_count of this GarbageCollectionDTO. The number of times garbage collection has run.

**Returns** The collection\_count of this GarbageCollectionDTO.

**Return type** int

**collection\_millis**

Gets the collection\_millis of this GarbageCollectionDTO. The total number of milliseconds spent garbage collecting.

**Returns** The collection\_millis of this GarbageCollectionDTO.

**Return type** int

**collection\_time**

Gets the collection\_time of this GarbageCollectionDTO. The total amount of time spent garbage collecting.

**Returns** The collection\_time of this GarbageCollectionDTO.

**Return type** str

**name**

Gets the name of this GarbageCollectionDTO. The name of the garbage collector.

**Returns** The name of this GarbageCollectionDTO.

**Return type** str

**swagger\_types** = {'collection\_count': 'int', 'collection\_millis': 'int', 'collection\_time': 'int', 'name': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.history\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.history\_dto.**HistoryDTO** (*total=None, last\_refreshed=None, actions=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

HistoryDTO - a model defined in Swagger

**actions**

Gets the actions of this HistoryDTO. The actions.

**Returns** The actions of this HistoryDTO.

**Return type** list[*ActionEntity*]

**attribute\_map** = {'actions': 'actions', 'last\_refreshed': 'lastRefreshed', 'total': 'total'}

**last\_refreshed**

Gets the last\_refreshed of this HistoryDTO. The timestamp when the report was generated.

**Returns** The last\_refreshed of this HistoryDTO.

**Return type** str

**swagger\_types** = {'actions': 'list[ActionEntity]', 'last\_refreshed': 'str', 'total':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**total**

Gets the total of this HistoryDTO. The number of number of actions that matched the search criteria..

**Returns** The total of this HistoryDTO.

**Return type** int

## nipyapi.nifi.models.history\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.history\_entity.**HistoryEntity** (*history=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

HistoryEntity - a model defined in Swagger

**attribute\_map** = {'history': 'history'}

**history**

Gets the history of this HistoryEntity.

**Returns** The history of this HistoryEntity.

**Return type** *HistoryDTO*

**swagger\_types** = {'history': 'HistoryDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.input\_ports\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.input_ports_entity.InputPortsEntity (input_ports=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

InputPortsEntity - a model defined in Swagger

```
attribute_map = {'input_ports': 'inputPorts'}
```

```
input_ports
```

Gets the input\_ports of this InputPortsEntity.

**Returns** The input\_ports of this InputPortsEntity.

**Return type** list[PortEntity]

```
swagger_types = {'input_ports': 'list[PortEntity]'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

### nipyapi.nifi.models.instantiate\_template\_request\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

InstantiateTemplateRequestEntity - a model defined in Swagger

```
attribute_map = {'disconnected_node_acknowledged': 'disconnectedNodeAcknowledged', 'e
```

```
disconnected_node_acknowledged
```

Gets the disconnected\_node\_acknowledged of this InstantiateTemplateRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The `disconnected_node_acknowledged` of this `InstantiateTemplateRequestEntity`.

**Return type** `bool`

**encoding\_version**

Gets the `encoding_version` of this `InstantiateTemplateRequestEntity`. The encoding version of the flow snippet. If not specified, this is automatically populated by the node receiving the user request. If the snippet is specified, the version will be the latest. If the snippet is not specified, the version will come from the underlying template. These details need to be replicated throughout the cluster to ensure consistency.

**Returns** The `encoding_version` of this `InstantiateTemplateRequestEntity`.

**Return type** `str`

**origin\_x**

Gets the `origin_x` of this `InstantiateTemplateRequestEntity`. The x coordinate of the origin of the bounding box where the new components will be placed.

**Returns** The `origin_x` of this `InstantiateTemplateRequestEntity`.

**Return type** `float`

**origin\_y**

Gets the `origin_y` of this `InstantiateTemplateRequestEntity`. The y coordinate of the origin of the bounding box where the new components will be placed.

**Returns** The `origin_y` of this `InstantiateTemplateRequestEntity`.

**Return type** `float`

**snippet**

Gets the snippet of this `InstantiateTemplateRequestEntity`. A flow snippet of the template contents. If not specified, this is automatically populated by the node receiving the user request. These details need to be replicated throughout the cluster to ensure consistency.

**Returns** The snippet of this `InstantiateTemplateRequestEntity`.

**Return type** *FlowSnippetDTO*

**swagger\_types** = {'disconnected\_node\_acknowledged': 'bool', 'encoding\_version': 'str'}

**template\_id**

Gets the `template_id` of this `InstantiateTemplateRequestEntity`. The identifier of the template.

**Returns** The `template_id` of this `InstantiateTemplateRequestEntity`.

**Return type** `str`

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.label\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.label_dto.LabelDTO (id=None, versioned_component_id=None,  
                                              parent_group_id=None, position=None,  
                                              label=None, width=None, height=None,  
                                              getz_index=None, style=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LabelDTO - a model defined in Swagger

```
attribute_map = {'getz_index':  'getzIndex', 'height':  'height', 'id':  'id', 'label':  'label'}
```

**getz\_index**

Gets the getz\_index of this LabelDTO. The z index of the label.

**Returns** The getz\_index of this LabelDTO.

**Return type** int

**height**

Gets the height of this LabelDTO. The height of the label in pixels when at a 1:1 scale.

**Returns** The height of this LabelDTO.

**Return type** float

**id**

Gets the id of this LabelDTO. The id of the component.

**Returns** The id of this LabelDTO.

**Return type** str

**label**

Gets the label of this LabelDTO. The text that appears in the label.

**Returns** The label of this LabelDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this LabelDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this LabelDTO.

**Return type** str

**position**

Gets the position of this LabelDTO. The position of this component in the UI if applicable.

**Returns** The position of this LabelDTO.

**Return type** *PositionDTO*

**style**

Gets the style of this LabelDTO. The styles for this label (font-size : 12px, background-color : #eee, etc).

**Returns** The style of this LabelDTO.

**Return type** dict(str, str)

```
swagger_types = {'getz_index':  'int', 'height':  'float', 'id':  'str', 'label':  'str'}
```

**to\_dict** ()

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this LabelDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this LabelDTO.

**Return type** str

**width**

Gets the width of this LabelDTO. The width of the label in pixels when at a 1:1 scale.

**Returns** The width of this LabelDTO.

**Return type** float

## nipyapi.nifi.models.label\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.label_entity.LabelEntity (revision=None,          id=None,
                                                    uri=None,          position=None,
                                                    permissions=None,      bul-
                                                    letins=None,          discon-
                                                    nected_node_acknowledged=None,
                                                    dimensions=None,
                                                    getz_index=None,      compo-
                                                    nent=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LabelEntity - a model defined in Swagger

```
attribute_map = {'bulletins':  'bulletins', 'component':  'component', 'dimensions':
```

**bulletins**

Gets the bulletins of this LabelEntity. The bulletins for this component.

**Returns** The bulletins of this LabelEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this LabelEntity.

**Returns** The component of this LabelEntity.

**Return type** *LabelDTO*

**dimensions**

Gets the dimensions of this LabelEntity.

**Returns** The dimensions of this LabelEntity.



**Return type** *DimensionsDTO*

#### **disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this LabelEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this LabelEntity.

**Return type** bool

#### **getz\_index**

Gets the getz\_index of this LabelEntity. The z index of the label.

**Returns** The getz\_index of this LabelEntity.

**Return type** int

#### **id**

Gets the id of this LabelEntity. The id of the component.

**Returns** The id of this LabelEntity.

**Return type** str

#### **permissions**

Gets the permissions of this LabelEntity. The permissions for this component.

**Returns** The permissions of this LabelEntity.

**Return type** *PermissionsDTO*

#### **position**

Gets the position of this LabelEntity. The position of this component in the UI if applicable.

**Returns** The position of this LabelEntity.

**Return type** *PositionDTO*

#### **revision**

Gets the revision of this LabelEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this LabelEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'LabelDTO', 'dime

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **uri**

Gets the uri of this LabelEntity. The URI for futures requests to the component.

**Returns** The uri of this LabelEntity.

**Return type** str

### **nipyapi.nifi.models.labels\_entity module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.labels_entity.LabelsEntity (labels=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LabelsEntity - a model defined in Swagger

```
attribute_map = {'labels': 'labels'}
```

```
labels
```

Gets the labels of this LabelsEntity.

**Returns** The labels of this LabelsEntity.

**Return type** list[[LabelEntity](#)]

```
swagger_types = {'labels': 'list[LabelEntity]'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

## **nipyapi.nifi.models.lineage\_dto module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.lineage_dto.LineageDTO (id=None, uri=None, submission_time=None, expiration=None, percent_completed=None, finished=None, request=None, results=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LineageDTO - a model defined in Swagger

```
attribute_map = {'expiration': 'expiration', 'finished': 'finished', 'id': 'id', 'p
```

```
expiration
```

Gets the expiration of this LineageDTO. When the lineage query will expire.

**Returns** The expiration of this LineageDTO.

**Return type** str

```
finished
```

Gets the finished of this LineageDTO. Whether the lineage query has finished.

**Returns** The finished of this LineageDTO.

**Return type** bool

**id**

Gets the id of this LineageDTO. The id of this lineage query.

**Returns** The id of this LineageDTO.

**Return type** str

**percent\_completed**

Gets the percent\_completed of this LineageDTO. The percent complete for the lineage query.

**Returns** The percent\_completed of this LineageDTO.

**Return type** int

**request**

Gets the request of this LineageDTO. The initial lineage result.

**Returns** The request of this LineageDTO.

**Return type** *LineageRequestDTO*

**results**

Gets the results of this LineageDTO. The results of the lineage query.

**Returns** The results of this LineageDTO.

**Return type** *LineageResultsDTO*

**submission\_time**

Gets the submission\_time of this LineageDTO. When the lineage query was submitted.

**Returns** The submission\_time of this LineageDTO.

**Return type** str

**swagger\_types** = {'expiration': 'str', 'finished': 'bool', 'id': 'str', 'percent\_com

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**uri**

Gets the uri of this LineageDTO. The URI for this lineage query for later retrieval and deletion.

**Returns** The uri of this LineageDTO.

**Return type** str

## nipyapi.nifi.models.lineage\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.lineage_entity.LineageEntity (lineage=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LineageEntity - a model defined in Swagger

```
attribute_map = {'lineage': 'lineage'}
```

```
lineage
```

Gets the lineage of this LineageEntity.

**Returns** The lineage of this LineageEntity.

**Return type** *LineageDTO*

```
swagger_types = {'lineage': 'LineageDTO'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

## nipyapi.nifi.models.lineage\_request\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.lineage_request_dto.LineageRequestDTO (event_id=None,
                                                                lin-
                                                                eage_request_type=None,
                                                                uuid=None,
                                                                clus-
                                                                ter_node_id=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LineageRequestDTO - a model defined in Swagger

```
attribute_map = {'cluster_node_id': 'clusterNodeId', 'event_id': 'eventId', 'lineage'
```

```
cluster_node_id
```

Gets the cluster\_node\_id of this LineageRequestDTO. The id of the node where this lineage originated if clustered.

**Returns** The cluster\_node\_id of this LineageRequestDTO.

**Return type** str

```
event_id
```

Gets the event\_id of this LineageRequestDTO. The event id that was used to generate this lineage, if applicable. The event id is allowed for any type of lineageRequestType. If the lineageRequestType is FLOWFILE and the flowfile uuid is also included in the request, the event id will be ignored.

**Returns** The event\_id of this LineageRequestDTO.

**Return type** int

#### **lineage\_request\_type**

Gets the lineage\_request\_type of this LineageRequestDTO. The type of lineage request. PARENTS will return the lineage for the flowfiles that are parents of the specified event. CHILDREN will return the lineage for the flowfiles that are children of the specified event. FLOWFILE will return the lineage for the specified flowfile.

**Returns** The lineage\_request\_type of this LineageRequestDTO.

**Return type** str

**swagger\_types** = {'cluster\_node\_id': 'str', 'event\_id': 'int', 'lineage\_request\_type':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

#### **uuid**

Gets the uuid of this LineageRequestDTO. The flowfile uuid that was used to generate the lineage. The flowfile uuid is only allowed when the lineageRequestType is FLOWFILE and will take precedence over event id.

**Returns** The uuid of this LineageRequestDTO.

**Return type** str

### **nipyapi.nifi.models.lineage\_results\_dto module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.lineage_results_dto.LineageResultsDTO (errors=None,
                                                                nodes=None,
                                                                links=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

LineageResultsDTO - a model defined in Swagger

**attribute\_map** = {'errors': 'errors', 'links': 'links', 'nodes': 'nodes'}

#### **errors**

Gets the errors of this LineageResultsDTO. Any errors that occurred while generating the lineage.

**Returns** The errors of this LineageResultsDTO.

**Return type** list[str]

#### **links**

Gets the links of this LineageResultsDTO. The links between the nodes in the lineage.

**Returns** The links of this LineageResultsDTO.

**Return type** list[*ProvenanceLinkDTO*]

**nodes**

Gets the nodes of this LineageResultsDTO. The nodes in the lineage.

**Returns** The nodes of this LineageResultsDTO.

**Return type** list[*ProvenanceNodeDTO*]

**swagger\_types** = {'errors': 'list[str]', 'links': 'list[ProvenanceLinkDTO]', 'nodes':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.listing\_request\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.listing_request_dto.ListingRequestDTO(id=None,
                                                                uri=None,
                                                                submis-
                                                                sion_time=None,
                                                                last_updated=None,
                                                                per-
                                                                cent_completed=None,
                                                                fin-
                                                                ished=None,
                                                                fail-
                                                                ure_reason=None,
                                                                max_results=None,
                                                                state=None,
                                                                queue_size=None,
                                                                flow_file_summaries=None,
                                                                source_running=None,
                                                                destina-
                                                                tion_running=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ListingRequestDTO - a model defined in Swagger

**attribute\_map** = {'destination\_running': 'destinationRunning', 'failure\_reason': 'fail-

**destination\_running**

Gets the destination\_running of this ListingRequestDTO. Whether the destination of the connection is running

**Returns** The destination\_running of this ListingRequestDTO.

**Return type** bool

**failure\_reason**

Gets the failure\_reason of this ListingRequestDTO. The reason, if any, that this listing request failed.

**Returns** The failure\_reason of this ListingRequestDTO.

**Return type** str

**finished**

Gets the finished of this ListingRequestDTO. Whether the query has finished.

**Returns** The finished of this ListingRequestDTO.

**Return type** bool

**flow\_file\_summaries**

Gets the flow\_file\_summaries of this ListingRequestDTO. The FlowFile summaries. The summaries will be populated once the request has completed.

**Returns** The flow\_file\_summaries of this ListingRequestDTO.

**Return type** list[[FlowFileSummaryDTO](#)]

**id**

Gets the id of this ListingRequestDTO. The id for this listing request.

**Returns** The id of this ListingRequestDTO.

**Return type** str

**last\_updated**

Gets the last\_updated of this ListingRequestDTO. The last time this listing request was updated.

**Returns** The last\_updated of this ListingRequestDTO.

**Return type** str

**max\_results**

Gets the max\_results of this ListingRequestDTO. The maximum number of FlowFileSummary objects to return

**Returns** The max\_results of this ListingRequestDTO.

**Return type** int

**percent\_completed**

Gets the percent\_completed of this ListingRequestDTO. The current percent complete.

**Returns** The percent\_completed of this ListingRequestDTO.

**Return type** int

**queue\_size**

Gets the queue\_size of this ListingRequestDTO. The size of the queue

**Returns** The queue\_size of this ListingRequestDTO.

**Return type** [QueueSizeDTO](#)

**source\_running**

Gets the source\_running of this ListingRequestDTO. Whether the source of the connection is running

**Returns** The source\_running of this ListingRequestDTO.

**Return type** bool

**state**

Gets the state of this ListingRequestDTO. The current state of the listing request.

**Returns** The state of this ListingRequestDTO.

**Return type** str

**submission\_time**

Gets the submission\_time of this ListingRequestDTO. The timestamp when the query was submitted.

**Returns** The submission\_time of this ListingRequestDTO.

**Return type** str

**swagger\_types** = {'destination\_running': 'bool', 'failure\_reason': 'str', 'finished':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**uri**

Gets the uri of this ListingRequestDTO. The URI for future requests to this listing request.

**Returns** The uri of this ListingRequestDTO.

**Return type** str

## nipyapi.nifi.models.listing\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.listing\_request\_entity.**ListingRequestEntity** (*listing\_request=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ListingRequestEntity - a model defined in Swagger

**attribute\_map** = {'listing\_request': 'listingRequest'}

**listing\_request**

Gets the listing\_request of this ListingRequestEntity.

**Returns** The listing\_request of this ListingRequestEntity.

**Return type** *ListingRequestDTO*

**swagger\_types** = {'listing\_request': 'ListingRequestDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model



## nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto.**NodeConnectionStatusSnapshotDTO**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeConnectionStatusSnapshotDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeConnectionStatusSnapshotDTO. The API address of the node

**Returns** The address of this NodeConnectionStatusSnapshotDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeConnectionStatusSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeConnectionStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

#### **node\_id**

Gets the node\_id of this NodeConnectionStatusSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeConnectionStatusSnapshotDTO.

**Return type** str

#### **status\_snapshot**

Gets the status\_snapshot of this NodeConnectionStatusSnapshotDTO. The connection status snapshot from the node.

**Returns** The status\_snapshot of this NodeConnectionStatusSnapshotDTO.

**Return type** *ConnectionStatusSnapshotDTO*

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'status\_sna

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_counters\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.node_counters_snapshot_dto.NodeCountersSnapshotDTO(node_id=None,
                                                                           ad-
                                                                           dress=None,
                                                                           api_port=None,
                                                                           snap-
                                                                           shot=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeCountersSnapshotDTO - a model defined in Swagger

#### address

Gets the address of this NodeCountersSnapshotDTO. The API address of the node

**Returns** The address of this NodeCountersSnapshotDTO.

**Return type** str

#### api\_port

Gets the api\_port of this NodeCountersSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeCountersSnapshotDTO.

**Return type** int

```
attribute_map = {'address': 'address', 'api_port': 'apiPort', 'node_id': 'nodeId',
```

#### node\_id

Gets the node\_id of this NodeCountersSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeCountersSnapshotDTO.

**Return type** str

#### snapshot

Gets the snapshot of this NodeCountersSnapshotDTO. The counters from the node.

**Returns** The snapshot of this NodeCountersSnapshotDTO.

**Return type** *CountersSnapshotDTO*

```
swagger_types = {'address': 'str', 'api_port': 'int', 'node_id': 'str', 'snapshot':
```

#### to\_dict()

Returns the model properties as a dict

#### to\_str()

Returns the string representation of the model

## nipyapi.nifi.models.node\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.node_dto.NodeDTO (node_id=None, address=None,
                                             api_port=None, status=None, heart-
                                             beat=None, connection_requested=None,
                                             roles=None, active_thread_count=None,
                                             queued=None, events=None,
                                             node_start_time=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeDTO - a model defined in Swagger

#### **active\_thread\_count**

Gets the active\_thread\_count of this NodeDTO. The active threads for the NiFi on the node.

**Returns** The active\_thread\_count of this NodeDTO.

**Return type** int

#### **address**

Gets the address of this NodeDTO. The node's host/ip address.

**Returns** The address of this NodeDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeDTO. The port the node is listening for API requests.

**Returns** The api\_port of this NodeDTO.

**Return type** int

**attribute\_map** = {'active\_thread\_count': 'activeThreadCount', 'address': 'address', 'api\_port': 'apiPort', 'connection\_requested': 'connectionRequested', 'events': 'events', 'heart-beat': 'heartBeat', 'node\_id': 'nodeId', 'node\_start\_time': 'nodeStartTime', 'queued': 'queued', 'roles': 'roles', 'status': 'status'}

#### **connection\_requested**

Gets the connection\_requested of this NodeDTO. The time of the node's last connection request.

**Returns** The connection\_requested of this NodeDTO.

**Return type** str

#### **events**

Gets the events of this NodeDTO. The node's events.

**Returns** The events of this NodeDTO.

**Return type** list[*NodeEventDTO*]

#### **heartbeat**

Gets the heartbeat of this NodeDTO. the time of the nodes's last heartbeat.

**Returns** The heartbeat of this NodeDTO.

**Return type** str

**node\_id**

Gets the node\_id of this NodeDTO. The id of the node.

**Returns** The node\_id of this NodeDTO.

**Return type** str

**node\_start\_time**

Gets the node\_start\_time of this NodeDTO. The time at which this Node was last refreshed.

**Returns** The node\_start\_time of this NodeDTO.

**Return type** str

**queued**

Gets the queued of this NodeDTO. The queue the NiFi on the node.

**Returns** The queued of this NodeDTO.

**Return type** str

**roles**

Gets the roles of this NodeDTO. The roles of this node.

**Returns** The roles of this NodeDTO.

**Return type** list[str]

**status**

Gets the status of this NodeDTO. The node's status.

**Returns** The status of this NodeDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'address': 'str', 'api\_port': 'int',

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.node\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_entity.**NodeEntity** (node=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeEntity - a model defined in Swagger

**attribute\_map** = {'node': 'node'}

**node**

Gets the node of this NodeEntity.

**Returns** The node of this NodeEntity.

**Return type** *NodeDTO*

**swagger\_types** = {'node': 'NodeDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_event\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_event\_dto.**NodeEventDTO**(*timestamp=None, category=None, message=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeEventDTO - a model defined in Swagger

**attribute\_map** = {'category': 'category', 'message': 'message', 'timestamp': 'timestamp'}

**category**

Gets the category of this NodeEventDTO. The category of the node event.

**Returns** The category of this NodeEventDTO.

**Return type** str

**message**

Gets the message of this NodeEventDTO. The message in the node event.

**Returns** The message of this NodeEventDTO.

**Return type** str

**swagger\_types** = {'category': 'str', 'message': 'str', 'timestamp': 'str'}

**timestamp**

Gets the timestamp of this NodeEventDTO. The timestamp of the node event.

**Returns** The timestamp of this NodeEventDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_port\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.node_port_status_snapshot_dto.NodePortStatusSnapshotDTO (node_id=None, address=None, api_port=None, status_snapshot=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodePortStatusSnapshotDTO - a model defined in Swagger

#### **address**

Gets the address of this NodePortStatusSnapshotDTO. The API address of the node

**Returns** The address of this NodePortStatusSnapshotDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodePortStatusSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodePortStatusSnapshotDTO.

**Return type** int

```
attribute_map = {'address': 'address', 'api_port': 'apiPort', 'node_id': 'nodeId',
```

#### **node\_id**

Gets the node\_id of this NodePortStatusSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodePortStatusSnapshotDTO.

**Return type** str

#### **status\_snapshot**

Gets the status\_snapshot of this NodePortStatusSnapshotDTO. The port status snapshot from the node.

**Returns** The status\_snapshot of this NodePortStatusSnapshotDTO.

**Return type** *PortStatusSnapshotDTO*

```
swagger_types = {'address': 'str', 'api_port': 'int', 'node_id': 'str', 'status_sna
```

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto.**NodeProcessGroupStatusSnapshotDTO**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeProcessGroupStatusSnapshotDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeProcessGroupStatusSnapshotDTO. The API address of the node

**Returns** The address of this NodeProcessGroupStatusSnapshotDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeProcessGroupStatusSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeProcessGroupStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

#### **node\_id**

Gets the node\_id of this NodeProcessGroupStatusSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeProcessGroupStatusSnapshotDTO.

**Return type** str

#### **status\_snapshot**

Gets the status\_snapshot of this NodeProcessGroupStatusSnapshotDTO. The process group status snapshot from the node.

**Returns** The status\_snapshot of this NodeProcessGroupStatusSnapshotDTO.

**Return type** *ProcessGroupStatusSnapshotDTO*

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'status\_sna

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.**NodeProcessorStatusSnapshotDTO**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeProcessorStatusSnapshotDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeProcessorStatusSnapshotDTO. The API address of the node

**Returns** The address of this NodeProcessorStatusSnapshotDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeProcessorStatusSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeProcessorStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

#### **node\_id**

Gets the node\_id of this NodeProcessorStatusSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeProcessorStatusSnapshotDTO.

**Return type** str

#### **status\_snapshot**

Gets the status\_snapshot of this NodeProcessorStatusSnapshotDTO. The processor status snapshot from the node.

**Returns** The status\_snapshot of this NodeProcessorStatusSnapshotDTO.

**Return type** *ProcessorStatusSnapshotDTO*

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'status\_sna

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model



## nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto.**NodeRemoteProcessG**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeRemoteProcessGroupStatusSnapshotDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeRemoteProcessGroupStatusSnapshotDTO. The API address of the node

**Returns** The address of this NodeRemoteProcessGroupStatusSnapshotDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeRemoteProcessGroupStatusSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeRemoteProcessGroupStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

#### **node\_id**

Gets the node\_id of this NodeRemoteProcessGroupStatusSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeRemoteProcessGroupStatusSnapshotDTO.

**Return type** str

#### **status\_snapshot**

Gets the status\_snapshot of this NodeRemoteProcessGroupStatusSnapshotDTO. The remote process group status snapshot from the node.

**Returns** The status\_snapshot of this NodeRemoteProcessGroupStatusSnapshotDTO.

**Return type** *RemoteProcessGroupStatusSnapshotDTO*

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'status\_sna

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.node\_search\_result\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.node_search_result_dto.NodeSearchResultDTO (id=None,  
                                                                    ad-  
                                                                    dress=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeSearchResultDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeSearchResultDTO. The address of the node that matched the search.

**Returns** The address of this NodeSearchResultDTO.

**Return type** str

```
attribute_map = {'address': 'address', 'id': 'id'}
```

#### **id**

Gets the id of this NodeSearchResultDTO. The id of the node that matched the search.

**Returns** The id of this NodeSearchResultDTO.

**Return type** str

```
swagger_types = {'address': 'str', 'id': 'str'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

## nipyapi.nifi.models.node\_status\_snapshots\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.node_status_snapshots_dto.NodeStatusSnapshotsDTO (node_id=None,  
                                                                    ad-  
                                                                    dress=None,  
                                                                    api_port=None,  
                                                                    sta-  
                                                                    tus_snapshots=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeStatusSnapshotsDTO - a model defined in Swagger

#### **address**

Gets the address of this NodeStatusSnapshotsDTO. The node's host/ip address.

**Returns** The address of this NodeStatusSnapshotsDTO.

**Return type** str

#### **api\_port**

Gets the api\_port of this NodeStatusSnapshotsDTO. The port the node is listening for API requests.

**Returns** The api\_port of this NodeStatusSnapshotsDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

#### **node\_id**

Gets the node\_id of this NodeStatusSnapshotsDTO. The id of the node.

**Returns** The node\_id of this NodeStatusSnapshotsDTO.

**Return type** str

#### **status\_snapshots**

Gets the status\_snapshots of this NodeStatusSnapshotsDTO. A list of StatusSnapshotDTO objects that provide the actual metric values for the component for this node.

**Returns** The status\_snapshots of this NodeStatusSnapshotsDTO.

**Return type** list[*StatusSnapshotDTO*]

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'status\_snap

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

### **nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto module**

#### **NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto.**NodeSystemDiagnosticsSnapshot**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

NodeSystemDiagnosticsSnapshotDTO - a model defined in Swagger

**address**

Gets the address of this NodeSystemDiagnosticsSnapshotDTO. The API address of the node

**Returns** The address of this NodeSystemDiagnosticsSnapshotDTO.

**Return type** str

**api\_port**

Gets the api\_port of this NodeSystemDiagnosticsSnapshotDTO. The API port used to communicate with the node

**Returns** The api\_port of this NodeSystemDiagnosticsSnapshotDTO.

**Return type** int

**attribute\_map** = {'address': 'address', 'api\_port': 'apiPort', 'node\_id': 'nodeId',

**node\_id**

Gets the node\_id of this NodeSystemDiagnosticsSnapshotDTO. The unique ID that identifies the node

**Returns** The node\_id of this NodeSystemDiagnosticsSnapshotDTO.

**Return type** str

**snapshot**

Gets the snapshot of this NodeSystemDiagnosticsSnapshotDTO. The System Diagnostics snapshot from the node.

**Returns** The snapshot of this NodeSystemDiagnosticsSnapshotDTO.

**Return type** *SystemDiagnosticsSnapshotDTO*

**swagger\_types** = {'address': 'str', 'api\_port': 'int', 'node\_id': 'str', 'snapshot':

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.output\_ports\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.output\_ports\_entity.**OutputPortsEntity** (output\_ports=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

OutputPortsEntity - a model defined in Swagger

**attribute\_map** = {'output\_ports': 'outputPorts'}

**output\_ports**

Gets the output\_ports of this OutputPortsEntity.

**Returns** The output\_ports of this OutputPortsEntity.

**Return type** list[PortEntity]

**swagger\_types** = {'output\_ports': 'list[PortEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.peer\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.peer\_dto.PeerDTO(*hostname=None, port=None, secure=None, flow\_file\_count=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PeerDTO - a model defined in Swagger

**attribute\_map** = {'flow\_file\_count': 'flowFileCount', 'hostname': 'hostname', 'port':

**flow\_file\_count**

Gets the flow\_file\_count of this PeerDTO. The number of flowFiles this peer holds.

**Returns** The flow\_file\_count of this PeerDTO.

**Return type** int

**hostname**

Gets the hostname of this PeerDTO. The hostname of this peer.

**Returns** The hostname of this PeerDTO.

**Return type** str

**port**

Gets the port of this PeerDTO. The port number of this peer.

**Returns** The port of this PeerDTO.

**Return type** int

**secure**

Gets the secure of this PeerDTO. Returns if this peer connection is secure.

**Returns** The secure of this PeerDTO.

**Return type** bool

**swagger\_types** = {'flow\_file\_count': 'int', 'hostname': 'str', 'port': 'int', 'secure

**to\_dict()**

Returns the model properties as a dict

```
to_str()
    Returns the string representation of the model
```

## nipyapi.nifi.models.peers\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.peers_entity.PeersEntity(peers=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PeersEntity - a model defined in Swagger

```
attribute_map = {'peers': 'peers'}
```

```
peers
```

Gets the peers of this PeersEntity.

**Returns** The peers of this PeersEntity.

**Return type** list[*PeerDTO*]

```
swagger_types = {'peers': 'list[PeerDTO]'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.permissions module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.permissions.Permissions(can_read=None, can_write=None,
                                                    can_delete=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Permissions - a model defined in Swagger

```
attribute_map = {'can_delete': 'canDelete', 'can_read': 'canRead', 'can_write': 'canWrite'}
```

```
can_delete
```

Gets the can\_delete of this Permissions. Indicates whether the user can delete a given resource.

**Returns** The can\_delete of this Permissions.

**Return type** bool

#### **can\_read**

Gets the can\_read of this Permissions. Indicates whether the user can read a given resource.

**Returns** The can\_read of this Permissions.

**Return type** bool

#### **can\_write**

Gets the can\_write of this Permissions. Indicates whether the user can write a given resource.

**Returns** The can\_write of this Permissions.

**Return type** bool

**swagger\_types** = {'can\_delete': 'bool', 'can\_read': 'bool', 'can\_write': 'bool'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

### nipyapi.nifi.models.permissions\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.permissions_dto.PermissionsDTO(can_read=None,
                                                         can_write=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PermissionsDTO - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'can\_write': 'canWrite'}

#### **can\_read**

Gets the can\_read of this PermissionsDTO. Indicates whether the user can read a given resource.

**Returns** The can\_read of this PermissionsDTO.

**Return type** bool

#### **can\_write**

Gets the can\_write of this PermissionsDTO. Indicates whether the user can write a given resource.

**Returns** The can\_write of this PermissionsDTO.

**Return type** bool

**swagger\_types** = {'can\_read': 'bool', 'can\_write': 'bool'}

#### **to\_dict()**

Returns the model properties as a dict

`to_str()`

Returns the string representation of the model

## nipyapi.nifi.models.port\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.port_dto.PortDTO(id=None, versioned_component_id=None,
                                           parent_group_id=None, position=None,
                                           name=None, comments=None, state=None,
                                           type=None, transmitting=None, concurrently_schedulable_task_count=None,
                                           user_access_control=None,
                                           group_access_control=None, allow_remote_access=None, validation_errors=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortDTO - a model defined in Swagger

#### **allow\_remote\_access**

Gets the allow\_remote\_access of this PortDTO. Whether this port can be accessed remotely via Site-to-Site protocol.

**Returns** The allow\_remote\_access of this PortDTO.

**Return type** bool

**attribute\_map** = {'allow\_remote\_access': 'allowRemoteAccess', 'comments': 'comments',

#### **comments**

Gets the comments of this PortDTO. The comments for the port.

**Returns** The comments of this PortDTO.

**Return type** str

#### **concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this PortDTO. The number of tasks that should be concurrently scheduled for the port.

**Returns** The concurrently\_schedulable\_task\_count of this PortDTO.

**Return type** int

#### **group\_access\_control**

Gets the group\_access\_control of this PortDTO. The user groups that are allowed to access the port.

**Returns** The group\_access\_control of this PortDTO.

**Return type** list[str]

#### **id**

Gets the id of this PortDTO. The id of the component.



**Returns** The id of this PortDTO.

**Return type** str

#### **name**

Gets the name of this PortDTO. The name of the port.

**Returns** The name of this PortDTO.

**Return type** str

#### **parent\_group\_id**

Gets the parent\_group\_id of this PortDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this PortDTO.

**Return type** str

#### **position**

Gets the position of this PortDTO. The position of this component in the UI if applicable.

**Returns** The position of this PortDTO.

**Return type** *PositionDTO*

#### **state**

Gets the state of this PortDTO. The state of the port.

**Returns** The state of this PortDTO.

**Return type** str

**swagger\_types** = {'allow\_remote\_access': 'bool', 'comments': 'str', 'concurrently\_sch

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **transmitting**

Gets the transmitting of this PortDTO. Whether the port has incoming or output connections to a remote NiFi. This is only applicable when the port is allowed to be accessed remotely.

**Returns** The transmitting of this PortDTO.

**Return type** bool

#### **type**

Gets the type of this PortDTO. The type of port.

**Returns** The type of this PortDTO.

**Return type** str

#### **user\_access\_control**

Gets the user\_access\_control of this PortDTO. The users that are allowed to access the port.

**Returns** The user\_access\_control of this PortDTO.

**Return type** list[str]

#### **validation\_errors**

Gets the validation\_errors of this PortDTO. Gets the validation errors from this port. These validation errors represent the problems with the port that must be resolved before it can be started.

**Returns** The validation\_errors of this PortDTO.

**Return type** list[str]

**versioned\_component\_id**

Gets the versioned\_component\_id of this PortDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this PortDTO.

**Return type** str

## nipyapi.nifi.models.port\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.port_entity.PortEntity (revision=None, id=None,
                                                    uri=None, position=None, permis-
                                                    sions=None, bulletins=None, discon-
                                                    nected_node_acknowledged=None,
                                                    component=None, sta-
                                                    tus=None, port_type=None,
                                                    operate_permissions=None, al-
                                                    low_remote_access=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortEntity - a model defined in Swagger

**allow\_remote\_access**

Gets the allow\_remote\_access of this PortEntity. Whether this port can be accessed remotely via Site-to-Site protocol.

**Returns** The allow\_remote\_access of this PortEntity.

**Return type** bool

```
attribute_map = {'allow_remote_access': 'allowRemoteAccess', 'bulletins': 'bulletins'}
```

**bulletins**

Gets the bulletins of this PortEntity. The bulletins for this component.

**Returns** The bulletins of this PortEntity.

**Return type** list[BulletinEntity]

**component**

Gets the component of this PortEntity.

**Returns** The component of this PortEntity.

**Return type** PortDTO

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this PortEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this PortEntity.

**Return type** bool

#### id

Gets the id of this PortEntity. The id of the component.

**Returns** The id of this PortEntity.

**Return type** str

#### operate\_permissions

Gets the operate\_permissions of this PortEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this PortEntity.

**Return type** *PermissionsDTO*

#### permissions

Gets the permissions of this PortEntity. The permissions for this component.

**Returns** The permissions of this PortEntity.

**Return type** *PermissionsDTO*

#### port\_type

Gets the port\_type of this PortEntity.

**Returns** The port\_type of this PortEntity.

**Return type** str

#### position

Gets the position of this PortEntity. The position of this component in the UI if applicable.

**Returns** The position of this PortEntity.

**Return type** *PositionDTO*

#### revision

Gets the revision of this PortEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this PortEntity.

**Return type** *RevisionDTO*

#### status

Gets the status of this PortEntity. The status of the port.

**Returns** The status of this PortEntity.

**Return type** *PortStatusDTO*

**swagger\_types** = {'allow\_remote\_access': 'bool', 'bulletins': 'list[BulletinEntity]',

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

#### uri

Gets the uri of this PortEntity. The URI for futures requests to the component.

**Returns** The uri of this PortEntity.

**Return type** str

## nipyapi.nifi.models.port\_status\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.port_status_dto.PortStatusDTO (id=None, group_id=None,
                                                         name=None,          trans-
                                                         mitting=None,
                                                         run_status=None,
                                                         stats_last_refreshed=None,
                                                         aggre-
                                                         gate_snapshot=None,
                                                         node_snapshots=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortStatusDTO - a model defined in Swagger

#### aggregate\_snapshot

Gets the aggregate\_snapshot of this PortStatusDTO. A status snapshot that represents the aggregate stats of all nodes in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this represents the stats of the single instance.

**Returns** The aggregate\_snapshot of this PortStatusDTO.

**Return type** *PortStatusSnapshotDTO*

```
attribute_map = {'aggregate_snapshot':  'aggregateSnapshot', 'group_id':  'groupId', 'id':  'id', 'name':  'name', 'node_snapshots':  'nodeSnapshots', 'run_status':  'runStatus', 'stats_last_refreshed':  'statsLastRefreshed', 'transmitting':  'transmitting'}
```

#### group\_id

Gets the group\_id of this PortStatusDTO. The id of the parent process group of the port.

**Returns** The group\_id of this PortStatusDTO.

**Return type** str

#### id

Gets the id of this PortStatusDTO. The id of the port.

**Returns** The id of this PortStatusDTO.

**Return type** str

#### name

Gets the name of this PortStatusDTO. The name of the port.

**Returns** The name of this PortStatusDTO.

**Return type** str

#### node\_snapshots

Gets the node\_snapshots of this PortStatusDTO. A status snapshot for each node in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this may be null.

**Returns** The node\_snapshots of this PortStatusDTO.

**Return type** list[*NodePortStatusSnapshotDTO*]

**run\_status**

Gets the run\_status of this PortStatusDTO. The run status of the port.

**Returns** The run\_status of this PortStatusDTO.

**Return type** str

**stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this PortStatusDTO. The time the status for the process group was last refreshed.

**Returns** The stats\_last\_refreshed of this PortStatusDTO.

**Return type** str

**swagger\_types** = {'aggregate\_snapshot': 'PortStatusSnapshotDTO', 'group\_id': 'str', '...

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmitting**

Gets the transmitting of this PortStatusDTO. Whether the port has incoming or outgoing connections to a remote NiFi.

**Returns** The transmitting of this PortStatusDTO.

**Return type** bool

**nipyapi.nifi.models.port\_status\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.port\_status\_entity.**PortStatusEntity**(port\_status=None, can\_read=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortStatusEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'port\_status': 'portStatus'}

**can\_read**

Gets the can\_read of this PortStatusEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this PortStatusEntity.

**Return type** bool

**port\_status**

Gets the port\_status of this PortStatusEntity.

**Returns** The port\_status of this PortStatusEntity.

**Return type** *PortStatusDTO*

```
swagger_types = {'can_read': 'bool', 'port_status': 'PortStatusDTO'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.port\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO(id=None,
                                                                           group_id=None,
                                                                           name=None,
                                                                           ac-
                                                                           tive_thread_count=None,
                                                                           flow_files_in=None,
                                                                           bytes_in=None,
                                                                           in-
                                                                           put=None,
                                                                           flow_files_out=None,
                                                                           bytes_out=None,
                                                                           out-
                                                                           put=None,
                                                                           trans-
                                                                           mit-
                                                                           ting=None,
                                                                           run_status=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortStatusSnapshotDTO - a model defined in Swagger

#### **active\_thread\_count**

Gets the active\_thread\_count of this PortStatusSnapshotDTO. The active thread count for the port.

**Returns** The active\_thread\_count of this PortStatusSnapshotDTO.

**Return type** int

```
attribute_map = {'active_thread_count': 'activeThreadCount', 'bytes_in': 'bytesIn',
```

#### **bytes\_in**

Gets the bytes\_in of this PortStatusSnapshotDTO. The size of hte FlowFiles that have been accepted in the last 5 minutes.

**Returns** The bytes\_in of this PortStatusSnapshotDTO.

**Return type** int

**bytes\_out**

Gets the bytes\_out of this PortStatusSnapshotDTO. The number of bytes that have been processed in the last 5 minutes.

**Returns** The bytes\_out of this PortStatusSnapshotDTO.

**Return type** int

**flow\_files\_in**

Gets the flow\_files\_in of this PortStatusSnapshotDTO. The number of FlowFiles that have been accepted in the last 5 minutes.

**Returns** The flow\_files\_in of this PortStatusSnapshotDTO.

**Return type** int

**flow\_files\_out**

Gets the flow\_files\_out of this PortStatusSnapshotDTO. The number of FlowFiles that have been processed in the last 5 minutes.

**Returns** The flow\_files\_out of this PortStatusSnapshotDTO.

**Return type** int

**group\_id**

Gets the group\_id of this PortStatusSnapshotDTO. The id of the parent process group of the port.

**Returns** The group\_id of this PortStatusSnapshotDTO.

**Return type** str

**id**

Gets the id of this PortStatusSnapshotDTO. The id of the port.

**Returns** The id of this PortStatusSnapshotDTO.

**Return type** str

**input**

Gets the input of this PortStatusSnapshotDTO. The count/size of flowfiles that have been accepted in the last 5 minutes.

**Returns** The input of this PortStatusSnapshotDTO.

**Return type** str

**name**

Gets the name of this PortStatusSnapshotDTO. The name of the port.

**Returns** The name of this PortStatusSnapshotDTO.

**Return type** str

**output**

Gets the output of this PortStatusSnapshotDTO. The count/size of flowfiles that have been processed in the last 5 minutes.

**Returns** The output of this PortStatusSnapshotDTO.

**Return type** str

**run\_status**

Gets the run\_status of this PortStatusSnapshotDTO. The run status of the port.

**Returns** The run\_status of this PortStatusSnapshotDTO.

**Return type** str

```
swagger_types = {'active_thread_count': 'int', 'bytes_in': 'int', 'bytes_out': 'int'
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
transmitting
    Gets the transmitting of this PortStatusSnapshotDTO. Whether the port has incoming or outgoing connections to a remote NiFi.

    Returns The transmitting of this PortStatusSnapshotDTO.

    Return type bool
```

### nipyapi.nifi.models.port\_status\_snapshot\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.port_status_snapshot_entity.PortStatusSnapshotEntity(id=None,
                                                                              port_status_snapshot=None,
                                                                              can_read=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PortStatusSnapshotEntity - a model defined in Swagger

attribute_map = {'can_read': 'canRead', 'id': 'id', 'port_status_snapshot': 'portStatusSnapshot'}

can_read
    Gets the can_read of this PortStatusSnapshotEntity. Indicates whether the user can read a given resource.

    Returns The can_read of this PortStatusSnapshotEntity.

    Return type bool

id
    Gets the id of this PortStatusSnapshotEntity. The id of the port.

    Returns The id of this PortStatusSnapshotEntity.

    Return type str

port_status_snapshot
    Gets the port_status_snapshot of this PortStatusSnapshotEntity.

    Returns The port_status_snapshot of this PortStatusSnapshotEntity.

    Return type PortStatusSnapshotDTO

swagger_types = {'can_read': 'bool', 'id': 'str', 'port_status_snapshot': 'PortStatusSnapshotDTO'}

to_dict()
    Returns the model properties as a dict
```



```
to_str()
```

Returns the string representation of the model

### nipyapi.nifi.models.position\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.position_dto.PositionDTO (x=None, y=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PositionDTO - a model defined in Swagger

```
attribute_map = {'x': 'x', 'y': 'y'}
```

```
swagger_types = {'x': 'float', 'y': 'float'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

**x**

Gets the x of this PositionDTO. The x coordinate.

**Returns** The x of this PositionDTO.

**Return type** float

**y**

Gets the y of this PositionDTO. The y coordinate.

**Returns** The y of this PositionDTO.

**Return type** float

### nipyapi.nifi.models.previous\_value\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.previous_value_dto.PreviousValueDTO (previous_value=None,
                                                                timestamp=None,
                                                                user_identity=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PreviousValueDTO - a model defined in Swagger

```
attribute_map = {'previous_value': 'previousValue', 'timestamp': 'timestamp', 'user_
```

**previous\_value**

Gets the previous\_value of this PreviousValueDTO. The previous value.

**Returns** The previous\_value of this PreviousValueDTO.

**Return type** str

```
swagger_types = {'previous_value': 'str', 'timestamp': 'str', 'user_identity': 'str
```

**timestamp**

Gets the timestamp of this PreviousValueDTO. The timestamp when the value was modified.

**Returns** The timestamp of this PreviousValueDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**user\_identity**

Gets the user\_identity of this PreviousValueDTO. The user who changed the previous value.

**Returns** The user\_identity of this PreviousValueDTO.

**Return type** str

## nipyapi.nifi.models.prioritizer\_types\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.prioritizer_types_entity.PrioritizerTypesEntity(prioritizer_types=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PrioritizerTypesEntity - a model defined in Swagger

```
attribute_map = {'prioritizer_types': 'prioritizerTypes'}
```

**prioritizer\_types**

Gets the prioritizer\_types of this PrioritizerTypesEntity.

**Returns** The prioritizer\_types of this PrioritizerTypesEntity.

**Return type** list[DocumentedTypeDTO]

```
swagger_types = {'prioritizer_types': 'list[DocumentedTypeDTO]'}
```

**to\_dict()**

Returns the model properties as a dict

`to_str()`

Returns the string representation of the model

### **nipyapi.nifi.models.process\_group\_dto module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.process_group_dto.ProcessGroupDTO (id=None, versioned_component_id=None, parent_group_id=None, position=None, name=None, comments=None, variables=None, version_control_information=None, parameter_context=None, flow_file_concurrency=None, flow_file_outbound_policy=None, default_fault_flow_file_expiration=None, default_fault_back_pressure_object_threshold=None, default_fault_back_pressure_data_size_threshold=None, running_count=None, stopped_count=None, invalid_count=None, disabled_count=None, active_remote_port_count=None, inactive_remote_port_count=None, up_to_date_count=None, locally_modified_count=None, stale_count=None, locally_modified_and_stale_count=None, sync_failure_count=None, local_input_port_count=None, local_output_port_count=None, public_input_port_count=None, public_output_port_count=None, contents=None, input_port_count=None, output_port_count=None)

```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupDTO - a model defined in Swagger

**active\_remote\_port\_count**

Gets the active\_remote\_port\_count of this ProcessGroupDTO. The number of active remote ports in the process group.

**Returns** The active\_remote\_port\_count of this ProcessGroupDTO.

**Return type** int

**attribute\_map** = {'active\_remote\_port\_count': 'activeRemotePortCount', 'comments': 'c

**comments**

Gets the comments of this ProcessGroupDTO. The comments for the process group.

**Returns** The comments of this ProcessGroupDTO.

**Return type** str

**contents**

Gets the contents of this ProcessGroupDTO. The contents of this process group.

**Returns** The contents of this ProcessGroupDTO.

**Return type** *FlowSnippetDTO*

**default\_back\_pressure\_data\_size\_threshold**

Gets the default\_back\_pressure\_data\_size\_threshold of this ProcessGroupDTO. Default value used in this Process Group for the maximum data size of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_data\_size\_threshold of this ProcessGroupDTO.

**Return type** str

**default\_back\_pressure\_object\_threshold**

Gets the default\_back\_pressure\_object\_threshold of this ProcessGroupDTO. Default value used in this Process Group for the maximum number of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_object\_threshold of this ProcessGroupDTO.

**Return type** int

**default\_flow\_file\_expiration**

Gets the default\_flow\_file\_expiration of this ProcessGroupDTO. The default FlowFile Expiration for this Process Group.

**Returns** The default\_flow\_file\_expiration of this ProcessGroupDTO.

**Return type** str

**disabled\_count**

Gets the disabled\_count of this ProcessGroupDTO. The number of disabled components in the process group.

**Returns** The disabled\_count of this ProcessGroupDTO.

**Return type** int

**flowfile\_concurrency**

Gets the flowfile\_concurrency of this ProcessGroupDTO. The FlowFile Concurrency for this Process Group.

**Returns** The flowfile\_concurrency of this ProcessGroupDTO.

**Return type** str

**flowfile\_outbound\_policy**

Gets the flowfile\_outbound\_policy of this ProcessGroupDTO. The Outbound Policy that is used for determining how FlowFiles should be transferred out of the Process Group.

**Returns** The flowfile\_outbound\_policy of this ProcessGroupDTO.

**Return type** str

**id**

Gets the id of this ProcessGroupDTO. The id of the component.

**Returns** The id of this ProcessGroupDTO.

**Return type** str

**inactive\_remote\_port\_count**

Gets the inactive\_remote\_port\_count of this ProcessGroupDTO. The number of inactive remote ports in the process group.

**Returns** The inactive\_remote\_port\_count of this ProcessGroupDTO.

**Return type** int

**input\_port\_count**

Gets the input\_port\_count of this ProcessGroupDTO. The number of input ports in the process group.

**Returns** The input\_port\_count of this ProcessGroupDTO.

**Return type** int

**invalid\_count**

Gets the invalid\_count of this ProcessGroupDTO. The number of invalid components in the process group.

**Returns** The invalid\_count of this ProcessGroupDTO.

**Return type** int

**local\_input\_port\_count**

Gets the local\_input\_port\_count of this ProcessGroupDTO. The number of local input ports in the process group.

**Returns** The local\_input\_port\_count of this ProcessGroupDTO.

**Return type** int

**local\_output\_port\_count**

Gets the local\_output\_port\_count of this ProcessGroupDTO. The number of local output ports in the process group.

**Returns** The local\_output\_port\_count of this ProcessGroupDTO.

**Return type** int

**locally\_modified\_and\_stale\_count**

Gets the locally\_modified\_and\_stale\_count of this ProcessGroupDTO. The number of locally modified and stale versioned process groups in the process group.

**Returns** The locally\_modified\_and\_stale\_count of this ProcessGroupDTO.

**Return type** int

**locally\_modified\_count**

Gets the locally\_modified\_count of this ProcessGroupDTO. The number of locally modified versioned process groups in the process group.

**Returns** The locally\_modified\_count of this ProcessGroupDTO.

**Return type** int

**name**

Gets the name of this ProcessGroupDTO. The name of the process group.

**Returns** The name of this ProcessGroupDTO.

**Return type** str

**output\_port\_count**

Gets the output\_port\_count of this ProcessGroupDTO. The number of output ports in the process group.

**Returns** The output\_port\_count of this ProcessGroupDTO.

**Return type** int

**parameter\_context**

Gets the parameter\_context of this ProcessGroupDTO. The Parameter Context that this Process Group is bound to.

**Returns** The parameter\_context of this ProcessGroupDTO.

**Return type** ParameterContextReferenceEntity

**parent\_group\_id**

Gets the parent\_group\_id of this ProcessGroupDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ProcessGroupDTO.

**Return type** str

**position**

Gets the position of this ProcessGroupDTO. The position of this component in the UI if applicable.

**Returns** The position of this ProcessGroupDTO.

**Return type** *PositionDTO*

**public\_input\_port\_count**

Gets the public\_input\_port\_count of this ProcessGroupDTO. The number of public input ports in the process group.

**Returns** The public\_input\_port\_count of this ProcessGroupDTO.

**Return type** int

**public\_output\_port\_count**

Gets the public\_output\_port\_count of this ProcessGroupDTO. The number of public output ports in the process group.

**Returns** The public\_output\_port\_count of this ProcessGroupDTO.

**Return type** int

**running\_count**

Gets the running\_count of this ProcessGroupDTO. The number of running components in this process group.

**Returns** The running\_count of this ProcessGroupDTO.

**Return type** int

**stale\_count**

Gets the stale\_count of this ProcessGroupDTO. The number of stale versioned process groups in the process group.

**Returns** The stale\_count of this ProcessGroupDTO.

**Return type** int

**stopped\_count**

Gets the stopped\_count of this ProcessGroupDTO. The number of stopped components in the process group.

**Returns** The stopped\_count of this ProcessGroupDTO.

**Return type** int

**swagger\_types** = {'active\_remote\_port\_count': 'int', 'comments': 'str', 'contents':

**sync\_failure\_count**

Gets the sync\_failure\_count of this ProcessGroupDTO. The number of versioned process groups in the process group that are unable to sync to a registry.

**Returns** The sync\_failure\_count of this ProcessGroupDTO.

**Return type** int

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**up\_to\_date\_count**

Gets the up\_to\_date\_count of this ProcessGroupDTO. The number of up to date versioned process groups in the process group.

**Returns** The up\_to\_date\_count of this ProcessGroupDTO.

**Return type** int

**variables**

Gets the variables of this ProcessGroupDTO. The variables that are configured for the Process Group. Note that this map contains only those variables that are defined on this Process Group and not any variables that are defined in the parent Process Group, etc. I.e., this Map will not contain all variables that are accessible by components in this Process Group by rather only the variables that are defined for this Process Group itself.

**Returns** The variables of this ProcessGroupDTO.

**Return type** dict(str, str)

**version\_control\_information**

Gets the version\_control\_information of this ProcessGroupDTO. The Version Control information that indicates which Flow Registry, and where in the Flow Registry, this Process Group is tracking to; or null if this Process Group is not under version control

**Returns** The version\_control\_information of this ProcessGroupDTO.

**Return type** *VersionControlInformationDTO*

**versioned\_component\_id**

Gets the versioned\_component\_id of this ProcessGroupDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ProcessGroupDTO.

**Return type** str



## nipyapi.nifi.models.process\_group\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_entity.ProcessGroupEntity (revision=None,
                                                                    id=None,
                                                                    uri=None,
                                                                    posi-
                                                                    tion=None,
                                                                    permis-
                                                                    sions=None,
                                                                    bul-
                                                                    letins=None,
                                                                    discon-
                                                                    nected_node_acknowledged=None,
                                                                    compo-
                                                                    nent=None,
                                                                    sta-
                                                                    tus=None,
                                                                    ver-
                                                                    sioned_flow_snapshot=None,
                                                                    run-
                                                                    ning_count=None,
                                                                    stopped_count=None,
                                                                    in-
                                                                    valid_count=None,
                                                                    dis-
                                                                    abled_count=None,
                                                                    ac-
                                                                    tive_remote_port_count=None,
                                                                    inac-
                                                                    tive_remote_port_count=None,
                                                                    ver-
                                                                    sioned_flow_state=None,
                                                                    up_to_date_count=None,
                                                                    lo-
                                                                    cally_modified_count=None,
                                                                    stale_count=None,
                                                                    lo-
                                                                    cally_modified_and_stale_count=None,
                                                                    sync_failure_count=None,
                                                                    lo-
                                                                    cal_input_port_count=None,
                                                                    lo-
                                                                    cal_output_port_count=None,
                                                                    pub-
                                                                    lic_input_port_count=None,
                                                                    pub-
                                                                    lic_output_port_count=None,
                                                                    parame-
                                                                    ter_context=None,
                                                                    in-
                                                                    put_port_count=None,
                                                                    out-
                                                                    put_port_count=None)
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupEntity - a model defined in Swagger

#### **active\_remote\_port\_count**

Gets the active\_remote\_port\_count of this ProcessGroupEntity. The number of active remote ports in the process group.

**Returns** The active\_remote\_port\_count of this ProcessGroupEntity.

**Return type** int

**attribute\_map** = {'active\_remote\_port\_count': 'activeRemotePortCount', 'bulletins': 'bulletins'}

#### **bulletins**

Gets the bulletins of this ProcessGroupEntity. The bulletins for this component.

**Returns** The bulletins of this ProcessGroupEntity.

**Return type** list[*BulletinEntity*]

#### **component**

Gets the component of this ProcessGroupEntity.

**Returns** The component of this ProcessGroupEntity.

**Return type** *ProcessGroupDTO*

#### **disabled\_count**

Gets the disabled\_count of this ProcessGroupEntity. The number of disabled components in the process group.

**Returns** The disabled\_count of this ProcessGroupEntity.

**Return type** int

#### **disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ProcessGroupEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ProcessGroupEntity.

**Return type** bool

#### **id**

Gets the id of this ProcessGroupEntity. The id of the component.

**Returns** The id of this ProcessGroupEntity.

**Return type** str

#### **inactive\_remote\_port\_count**

Gets the inactive\_remote\_port\_count of this ProcessGroupEntity. The number of inactive remote ports in the process group.

**Returns** The inactive\_remote\_port\_count of this ProcessGroupEntity.

**Return type** int

#### **input\_port\_count**

Gets the input\_port\_count of this ProcessGroupEntity. The number of input ports in the process group.

**Returns** The input\_port\_count of this ProcessGroupEntity.

**Return type** int

#### **invalid\_count**

Gets the invalid\_count of this ProcessGroupEntity. The number of invalid components in the process group.

**Returns** The `invalid_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`local_input_port_count`**

Gets the `local_input_port_count` of this `ProcessGroupEntity`. The number of local input ports in the process group.

**Returns** The `local_input_port_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`local_output_port_count`**

Gets the `local_output_port_count` of this `ProcessGroupEntity`. The number of local output ports in the process group.

**Returns** The `local_output_port_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`locally_modified_and_stale_count`**

Gets the `locally_modified_and_stale_count` of this `ProcessGroupEntity`. The number of locally modified and stale versioned process groups in the process group.

**Returns** The `locally_modified_and_stale_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`locally_modified_count`**

Gets the `locally_modified_count` of this `ProcessGroupEntity`. The number of locally modified versioned process groups in the process group.

**Returns** The `locally_modified_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`output_port_count`**

Gets the `output_port_count` of this `ProcessGroupEntity`. The number of output ports in the process group.

**Returns** The `output_port_count` of this `ProcessGroupEntity`.

**Return type** `int`

**`parameter_context`**

Gets the `parameter_context` of this `ProcessGroupEntity`. The Parameter Context, or null if no Parameter Context has been bound to the Process Group

**Returns** The `parameter_context` of this `ProcessGroupEntity`.

**Return type** `ParameterContextReferenceEntity`

**`permissions`**

Gets the permissions of this `ProcessGroupEntity`. The permissions for this component.

**Returns** The permissions of this `ProcessGroupEntity`.

**Return type** *PermissionsDTO*

**`position`**

Gets the position of this `ProcessGroupEntity`. The position of this component in the UI if applicable.

**Returns** The position of this `ProcessGroupEntity`.

**Return type** *PositionDTO*

**public\_input\_port\_count**

Gets the `public_input_port_count` of this `ProcessGroupEntity`. The number of public input ports in the process group.

**Returns** The `public_input_port_count` of this `ProcessGroupEntity`.

**Return type** `int`

**public\_output\_port\_count**

Gets the `public_output_port_count` of this `ProcessGroupEntity`. The number of public output ports in the process group.

**Returns** The `public_output_port_count` of this `ProcessGroupEntity`.

**Return type** `int`

**revision**

Gets the revision of this `ProcessGroupEntity`. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this `ProcessGroupEntity`.

**Return type** *RevisionDTO*

**running\_count**

Gets the `running_count` of this `ProcessGroupEntity`. The number of running components in this process group.

**Returns** The `running_count` of this `ProcessGroupEntity`.

**Return type** `int`

**stale\_count**

Gets the `stale_count` of this `ProcessGroupEntity`. The number of stale versioned process groups in the process group.

**Returns** The `stale_count` of this `ProcessGroupEntity`.

**Return type** `int`

**status**

Gets the status of this `ProcessGroupEntity`. The status of the process group.

**Returns** The status of this `ProcessGroupEntity`.

**Return type** *ProcessGroupStatusDTO*

**stopped\_count**

Gets the `stopped_count` of this `ProcessGroupEntity`. The number of stopped components in the process group.

**Returns** The `stopped_count` of this `ProcessGroupEntity`.

**Return type** `int`

**swagger\_types** = {'active\_remote\_port\_count': 'int', 'bulletins': 'list[BulletinEntity]'

**sync\_failure\_count**

Gets the `sync_failure_count` of this `ProcessGroupEntity`. The number of versioned process groups in the process group that are unable to sync to a registry.

**Returns** The `sync_failure_count` of this `ProcessGroupEntity`.

**Return type** `int`

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**up\_to\_date\_count**

Gets the up\_to\_date\_count of this ProcessGroupEntity. The number of up to date versioned process groups in the process group.

**Returns** The up\_to\_date\_count of this ProcessGroupEntity.

**Return type** int

**uri**

Gets the uri of this ProcessGroupEntity. The URI for futures requests to the component.

**Returns** The uri of this ProcessGroupEntity.

**Return type** str

**versioned\_flow\_snapshot**

Gets the versioned\_flow\_snapshot of this ProcessGroupEntity. Returns the Versioned Flow that describes the contents of the Versioned Flow to be imported

**Returns** The versioned\_flow\_snapshot of this ProcessGroupEntity.

**Return type** *VersionedFlowSnapshot*

**versioned\_flow\_state**

Gets the versioned\_flow\_state of this ProcessGroupEntity. The current state of the Process Group, as it relates to the Versioned Flow

**Returns** The versioned\_flow\_state of this ProcessGroupEntity.

**Return type** str

## nipyapi.nifi.models.process\_group\_flow\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_flow_dto.ProcessGroupFlowDTO (id=None,
                                                                    uri=None,
                                                                    par-
                                                                    ent_group_id=None,
                                                                    pa-
                                                                    rame-
                                                                    ter_context=None,
                                                                    bread-
                                                                    crumb=None,
                                                                    flow=None,
                                                                    last_refreshed=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupFlowDTO - a model defined in Swagger

**attribute\_map** = {'breadcrumb': 'breadcrumb', 'flow': 'flow', 'id': 'id', 'last\_refreshed': 'last\_refreshed'}

#### **breadcrumb**

Gets the breadcrumb of this ProcessGroupFlowDTO. The breadcrumb of the process group.

**Returns** The breadcrumb of this ProcessGroupFlowDTO.

**Return type** *FlowBreadcrumbEntity*

#### **flow**

Gets the flow of this ProcessGroupFlowDTO. The flow structure starting at this Process Group.

**Returns** The flow of this ProcessGroupFlowDTO.

**Return type** *FlowDTO*

#### **id**

Gets the id of this ProcessGroupFlowDTO. The id of the component.

**Returns** The id of this ProcessGroupFlowDTO.

**Return type** str

#### **last\_refreshed**

Gets the last\_refreshed of this ProcessGroupFlowDTO. The time the flow for the process group was last refreshed.

**Returns** The last\_refreshed of this ProcessGroupFlowDTO.

**Return type** str

#### **parameter\_context**

Gets the parameter\_context of this ProcessGroupFlowDTO. The Parameter Context, or null if no Parameter Context has been bound to the Process Group

**Returns** The parameter\_context of this ProcessGroupFlowDTO.

**Return type** ParameterContextReferenceEntity

#### **parent\_group\_id**

Gets the parent\_group\_id of this ProcessGroupFlowDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ProcessGroupFlowDTO.

**Return type** str

**swagger\_types** = {'breadcrumb': 'FlowBreadcrumbEntity', 'flow': 'FlowDTO', 'id': 'str', 'last\_refreshed': 'datetime', 'parameter\_context': 'ParameterContextReferenceEntity', 'parent\_group\_id': 'str'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **uri**

Gets the uri of this ProcessGroupFlowDTO. The URI for futures requests to the component.

**Returns** The uri of this ProcessGroupFlowDTO.

**Return type** str

## nipyapi.nifi.models.process\_group\_flow\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_flow_entity.ProcessGroupFlowEntity (permissions=None,  
process_group_flow=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupFlowEntity - a model defined in Swagger

```
attribute_map = {'permissions': 'permissions', 'process_group_flow': 'processGroupFlowEntityPermissions'}  
permissions
```

Gets the permissions of this ProcessGroupFlowEntity. The access policy for this process group.

**Returns** The permissions of this ProcessGroupFlowEntity.

**Return type** *PermissionsDTO*

```
process_group_flow
```

Gets the process\_group\_flow of this ProcessGroupFlowEntity.

**Returns** The process\_group\_flow of this ProcessGroupFlowEntity.

**Return type** *ProcessGroupFlowDTO*

```
swagger_types = {'permissions': 'PermissionsDTO', 'process_group_flow': 'ProcessGroupFlowEntity'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

## nipyapi.nifi.models.process\_group\_status\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_status_dto.ProcessGroupStatusDTO (id=None,  
name=None,  
stats_last_refreshed=None,  
aggregate_snapshot=None,  
gate_snapshot=None,  
node_snapshots=None)
```



Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupStatusDTO - a model defined in Swagger

#### **aggregate\_snapshot**

Gets the aggregate\_snapshot of this ProcessGroupStatusDTO. The aggregate status of all nodes in the cluster

**Returns** The aggregate\_snapshot of this ProcessGroupStatusDTO.

**Return type** *ProcessGroupStatusSnapshotDTO*

**attribute\_map** = {'aggregate\_snapshot': 'aggregateSnapshot', 'id': 'id', 'name': 'name'}

#### **id**

Gets the id of this ProcessGroupStatusDTO. The ID of the Process Group

**Returns** The id of this ProcessGroupStatusDTO.

**Return type** `str`

#### **name**

Gets the name of this ProcessGroupStatusDTO. The name of the Process Group

**Returns** The name of this ProcessGroupStatusDTO.

**Return type** `str`

#### **node\_snapshots**

Gets the node\_snapshots of this ProcessGroupStatusDTO. The status reported by each node in the cluster. If the NiFi instance is a standalone instance, rather than a clustered instance, this value may be null.

**Returns** The node\_snapshots of this ProcessGroupStatusDTO.

**Return type** `list[NodeProcessGroupStatusSnapshotDTO]`

#### **stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this ProcessGroupStatusDTO. The time the status for the process group was last refreshed.

**Returns** The stats\_last\_refreshed of this ProcessGroupStatusDTO.

**Return type** `str`

**swagger\_types** = {'aggregate\_snapshot': 'ProcessGroupStatusSnapshotDTO', 'id': 'str', 'name': 'str'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## **nipyapi.nifi.models.process\_group\_status\_entity module**

### **NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_status_entity.ProcessGroupStatusEntity (process_group_s,  
                                     can_read=None)  
    Bases: object  
  
    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
    ProcessGroupStatusEntity - a model defined in Swagger  
  
    attribute_map = {'can_read': 'canRead', 'process_group_status': 'processGroupStatus'}  
    can_read  
        Gets the can_read of this ProcessGroupStatusEntity. Indicates whether the user can read a given resource.  
        Returns The can_read of this ProcessGroupStatusEntity.  
        Return type bool  
  
    process_group_status  
        Gets the process_group_status of this ProcessGroupStatusEntity.  
        Returns The process_group_status of this ProcessGroupStatusEntity.  
        Return type ProcessGroupStatusDTO  
  
    swagger_types = {'can_read': 'bool', 'process_group_status': 'ProcessGroupStatusDTO'}  
    to_dict()  
        Returns the model properties as a dict  
    to_str()  
        Returns the string representation of the model
```

### **nipyapi.nifi.models.process\_group\_status\_snapshot\_dto module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO (i
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupStatusSnapshotDTO - a model defined in Swagger

**active\_thread\_count**

Gets the active\_thread\_count of this ProcessGroupStatusSnapshotDTO. The active thread count for this process group.

**Returns** The active\_thread\_count of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'active\_thread\_count': 'activeThreadCount', 'bytes\_in': 'bytesIn',

**bytes\_in**

Gets the bytes\_in of this ProcessGroupStatusSnapshotDTO. The number of bytes that have come into this ProcessGroup in the last 5 minutes

**Returns** The bytes\_in of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_out**

Gets the bytes\_out of this ProcessGroupStatusSnapshotDTO. The number of bytes transferred out of this ProcessGroup in the last 5 minutes

**Returns** The bytes\_out of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_queued**

Gets the bytes\_queued of this ProcessGroupStatusSnapshotDTO. The number of bytes that are queued up in this ProcessGroup right now

**Returns** The bytes\_queued of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_read**

Gets the bytes\_read of this ProcessGroupStatusSnapshotDTO. The number of bytes read by components in this ProcessGroup in the last 5 minutes

**Returns** The bytes\_read of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_received**

Gets the bytes\_received of this ProcessGroupStatusSnapshotDTO. The number of bytes received from external sources by components within this ProcessGroup in the last 5 minutes

**Returns** The bytes\_received of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_sent**

Gets the bytes\_sent of this ProcessGroupStatusSnapshotDTO. The number of bytes sent to an external sink by components within this ProcessGroup in the last 5 minutes

**Returns** The bytes\_sent of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_transferred**

Gets the bytes\_transferred of this ProcessGroupStatusSnapshotDTO. The number of bytes transferred in this ProcessGroup in the last 5 minutes

**Returns** The bytes\_transferred of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_written**

Gets the bytes\_written of this ProcessGroupStatusSnapshotDTO. The number of bytes written by components in this ProcessGroup in the last 5 minutes

**Returns** The bytes\_written of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**connection\_status\_snapshots**

Gets the connection\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all connections in the process group.

**Returns** The connection\_status\_snapshots of this ProcessGroupStatusSnapshotDTO.

**Return type** list[*ConnectionStatusSnapshotEntity*]

**flow\_files\_in**

Gets the flow\_files\_in of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles that have come into this ProcessGroup in the last 5 minutes

**Returns** The flow\_files\_in of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_out**

Gets the flow\_files\_out of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles transferred out of this ProcessGroup in the last 5 minutes

**Returns** The flow\_files\_out of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_queued**

Gets the flow\_files\_queued of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles that are queued up in this ProcessGroup right now

**Returns** The flow\_files\_queued of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_received**

Gets the flow\_files\_received of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles received from external sources by components within this ProcessGroup in the last 5 minutes

**Returns** The flow\_files\_received of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_sent**

Gets the flow\_files\_sent of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles sent to an external sink by components within this ProcessGroup in the last 5 minutes

**Returns** The flow\_files\_sent of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_transferred**

Gets the flow\_files\_transferred of this ProcessGroupStatusSnapshotDTO. The number of FlowFiles transferred in this ProcessGroup in the last 5 minutes

**Returns** The flow\_files\_transferred of this ProcessGroupStatusSnapshotDTO.

**Return type** int

**id**

Gets the id of this ProcessGroupStatusSnapshotDTO. The id of the process group.

**Returns** The id of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**input**

Gets the input of this ProcessGroupStatusSnapshotDTO. The input count/size for the process group in the last 5 minutes (pretty printed).

**Returns** The input of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**input\_port\_status\_snapshots**

Gets the input\_port\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all input ports in the process group.

**Returns** The input\_port\_status\_snapshots of this ProcessGroupStatusSnapshotDTO.

**Return type** list[[\*PortStatusSnapshotEntity\*](#)]

**name**

Gets the name of this ProcessGroupStatusSnapshotDTO. The name of this process group.

**Returns** The name of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**output**

Gets the output of this ProcessGroupStatusSnapshotDTO. The output count/size for the process group in the last 5 minutes.

**Returns** The output of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**output\_port\_status\_snapshots**

Gets the output\_port\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all output ports in the process group.

**Returns** The output\_port\_status\_snapshots of this ProcessGroupStatusSnapshotDTO.

**Return type** list[[\*PortStatusSnapshotEntity\*](#)]

**process\_group\_status\_snapshots**

Gets the process\_group\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all process groups in the process group.

**Returns** The process\_group\_status\_snapshots of this ProcessGroupStatusSnapshotDTO.

**Return type** list[[\*ProcessGroupStatusSnapshotEntity\*](#)]

**processor\_status\_snapshots**

Gets the processor\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all processors in the process group.

**Returns** The processor\_status\_snapshots of this ProcessGroupStatusSnapshotDTO.

**Return type** list[[\*ProcessorStatusSnapshotEntity\*](#)]

**queued**

Gets the queued of this ProcessGroupStatusSnapshotDTO. The count/size that is queued in the the process group.

**Returns** The queued of this ProcessGroupStatusSnapshotDTO.

**Return type** str

#### **queued\_count**

Gets the queued\_count of this ProcessGroupStatusSnapshotDTO. The count that is queued for the process group.

**Returns** The queued\_count of this ProcessGroupStatusSnapshotDTO.

**Return type** str

#### **queued\_size**

Gets the queued\_size of this ProcessGroupStatusSnapshotDTO. The size that is queued for the process group.

**Returns** The queued\_size of this ProcessGroupStatusSnapshotDTO.

**Return type** str

#### **read**

Gets the read of this ProcessGroupStatusSnapshotDTO. The number of bytes read in the last 5 minutes.

**Returns** The read of this ProcessGroupStatusSnapshotDTO.

**Return type** str

#### **received**

Gets the received of this ProcessGroupStatusSnapshotDTO. The count/size sent to the process group in the last 5 minutes.

**Returns** The received of this ProcessGroupStatusSnapshotDTO.

**Return type** str

#### **remote\_process\_group\_status\_snapshots**

Gets the remote\_process\_group\_status\_snapshots of this ProcessGroupStatusSnapshotDTO. The status of all remote process groups in the process group.

**Returns** The remote\_process\_group\_status\_snapshots of this ProcessGroupStatusSnapshot-DTO.

**Return type** list[*RemoteProcessGroupStatusSnapshotEntity*]

#### **sent**

Gets the sent of this ProcessGroupStatusSnapshotDTO. The count/size sent from this process group in the last 5 minutes.

**Returns** The sent of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'bytes\_in': 'int', 'bytes\_out': 'int'

#### **terminated\_thread\_count**

Gets the terminated\_thread\_count of this ProcessGroupStatusSnapshotDTO. The number of threads currently terminated for the process group.

**Returns** The terminated\_thread\_count of this ProcessGroupStatusSnapshotDTO.

**Return type** int

#### **to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transferred**

Gets the transferred of this ProcessGroupStatusSnapshotDTO. The count/size transferred to/from queues in the process group in the last 5 minutes.

**Returns** The transferred of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**versioned\_flow\_state**

Gets the versioned\_flow\_state of this ProcessGroupStatusSnapshotDTO. The current state of the Process Group, as it relates to the Versioned Flow

**Returns** The versioned\_flow\_state of this ProcessGroupStatusSnapshotDTO.

**Return type** str

**written**

Gets the written of this ProcessGroupStatusSnapshotDTO. The number of bytes written in the last 5 minutes.

**Returns** The written of this ProcessGroupStatusSnapshotDTO.

**Return type** str

## nipyapi.nifi.models.process\_group\_status\_snapshot\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.process\_group\_status\_snapshot\_entity.ProcessGroupStatusSnapshotEntity

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupStatusSnapshotEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'id': 'id', 'process\_group\_status\_snapshot':

**can\_read**

Gets the can\_read of this ProcessGroupStatusSnapshotEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this ProcessGroupStatusSnapshotEntity.

**Return type** bool

**id**

Gets the id of this ProcessGroupStatusSnapshotEntity. The id of the process group.

**Returns** The id of this ProcessGroupStatusSnapshotEntity.

**Return type** str



**process\_group\_status\_snapshot**

Gets the process\_group\_status\_snapshot of this ProcessGroupStatusSnapshotEntity.

**Returns** The process\_group\_status\_snapshot of this ProcessGroupStatusSnapshotEntity.

**Return type** *ProcessGroupStatusSnapshotDTO*

**swagger\_types** = {'can\_read': 'bool', 'id': 'str', 'process\_group\_status\_snapshot':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.process\_groups\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.process\_groups\_entity.**ProcessGroupsEntity**(process\_groups=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessGroupsEntity - a model defined in Swagger

**attribute\_map** = {'process\_groups': 'processGroups'}

**process\_groups**

Gets the process\_groups of this ProcessGroupsEntity.

**Returns** The process\_groups of this ProcessGroupsEntity.

**Return type** list[*ProcessGroupEntity*]

**swagger\_types** = {'process\_groups': 'list[ProcessGroupEntity]'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.processor\_config\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO (properties=None,
                                                                    descrip-
                                                                    tors=None,
                                                                    sensi-
                                                                    tive_dynamic_property_names=None,
                                                                    schedul-
                                                                    ing_period=None,
                                                                    schedul-
                                                                    ing_strategy=None,
                                                                    execu-
                                                                    tion_node=None,
                                                                    penalty_duration=None,
                                                                    yield_duration=None,
                                                                    bul-
                                                                    letin_level=None,
                                                                    run_duration_millis=None,
                                                                    concur-
                                                                    rently_schedulable_task_count=None,
                                                                    auto_terminated_relationships=None,
                                                                    com-
                                                                    ments=None,
                                                                    cus-
                                                                    tom_ui_url=None,
                                                                    loss_tolerant=None,
                                                                    annota-
                                                                    tion_data=None,
                                                                    de-
                                                                    fault_concurrent_tasks=None,
                                                                    de-
                                                                    fault_scheduling_period=None,
                                                                    retry_count=None,
                                                                    re-
                                                                    tried_relationships=None,
                                                                    back-
                                                                    off_mechanism=None,
                                                                    max_backoff_period=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorConfigDTO - a model defined in Swagger

#### **annotation\_data**

Gets the annotation\_data of this ProcessorConfigDTO. The annotation data for the processor used to relay configuration between a custom UI and the procesosr.

**Returns** The annotation\_data of this ProcessorConfigDTO.

**Return type** str

**attribute\_map** = {'annotation\_data': 'annotationData', 'auto\_terminated\_relationships':

#### **auto\_terminated\_relationships**

Gets the auto\_terminated\_relationships of this ProcessorConfigDTO. The names of all relationships that cause a flow file to be terminated if the relationship is not connected elsewhere. This property differs from the 'isAutoTerminate' property of the RelationshipDTO in that the RelationshipDTO is meant to depict the current configuration, whereas this property can be set in a DTO when updating a Processor in order to change which Relationships should be auto-terminated.

**Returns** The `auto_terminated_relationships` of this `ProcessorConfigDTO`.

**Return type** `list[str]`

**backoff\_mechanism**

Gets the `backoff_mechanism` of this `ProcessorConfigDTO`. Determines whether the `FlowFile` should be penalized or the processor should be yielded between retries.

**Returns** The `backoff_mechanism` of this `ProcessorConfigDTO`.

**Return type** `str`

**bulletin\_level**

Gets the `bulletin_level` of this `ProcessorConfigDTO`. The level at which the processor will report bulletins.

**Returns** The `bulletin_level` of this `ProcessorConfigDTO`.

**Return type** `str`

**comments**

Gets the comments of this `ProcessorConfigDTO`. The comments for the processor.

**Returns** The comments of this `ProcessorConfigDTO`.

**Return type** `str`

**concurrently\_schedulable\_task\_count**

Gets the `concurrently_schedulable_task_count` of this `ProcessorConfigDTO`. The number of tasks that should be concurrently schedule for the processor. If the processor doesn't allow parallel processing then any positive input will be ignored.

**Returns** The `concurrently_schedulable_task_count` of this `ProcessorConfigDTO`.

**Return type** `int`

**custom\_ui\_url**

Gets the `custom_ui_url` of this `ProcessorConfigDTO`. The URL for the processor's custom configuration UI if applicable.

**Returns** The `custom_ui_url` of this `ProcessorConfigDTO`.

**Return type** `str`

**default\_concurrent\_tasks**

Gets the `default_concurrent_tasks` of this `ProcessorConfigDTO`. Maps default values for concurrent tasks for each applicable scheduling strategy.

**Returns** The `default_concurrent_tasks` of this `ProcessorConfigDTO`.

**Return type** `dict(str, str)`

**default\_scheduling\_period**

Gets the `default_scheduling_period` of this `ProcessorConfigDTO`. Maps default values for scheduling period for each applicable scheduling strategy.

**Returns** The `default_scheduling_period` of this `ProcessorConfigDTO`.

**Return type** `dict(str, str)`

**descriptors**

Gets the descriptors of this `ProcessorConfigDTO`. Descriptors for the processor's properties.

**Returns** The descriptors of this `ProcessorConfigDTO`.

**Return type** `dict(str, PropertyDescriptorDTO)`

**execution\_node**

Gets the execution\_node of this ProcessorConfigDTO. Indicates the node where the process will execute.

**Returns** The execution\_node of this ProcessorConfigDTO.

**Return type** str

**loss\_tolerant**

Gets the loss\_tolerant of this ProcessorConfigDTO. Whether the processor is loss tolerant.

**Returns** The loss\_tolerant of this ProcessorConfigDTO.

**Return type** bool

**max\_backoff\_period**

Gets the max\_backoff\_period of this ProcessorConfigDTO. Maximum amount of time to be waited during a retry period.

**Returns** The max\_backoff\_period of this ProcessorConfigDTO.

**Return type** str

**penalty\_duration**

Gets the penalty\_duration of this ProcessorConfigDTO. The amount of time that is used when the process penalizes a flowfile.

**Returns** The penalty\_duration of this ProcessorConfigDTO.

**Return type** str

**properties**

Gets the properties of this ProcessorConfigDTO. The properties for the processor. Properties whose value is not set will only contain the property name.

**Returns** The properties of this ProcessorConfigDTO.

**Return type** dict(str, str)

**retried\_relationships**

Gets the retried\_relationships of this ProcessorConfigDTO. All the relationships should be retried.

**Returns** The retried\_relationships of this ProcessorConfigDTO.

**Return type** list[str]

**retry\_count**

Gets the retry\_count of this ProcessorConfigDTO. Overall number of retries.

**Returns** The retry\_count of this ProcessorConfigDTO.

**Return type** int

**run\_duration\_millis**

Gets the run\_duration\_millis of this ProcessorConfigDTO. The run duration for the processor in milliseconds.

**Returns** The run\_duration\_millis of this ProcessorConfigDTO.

**Return type** int

**scheduling\_period**

Gets the scheduling\_period of this ProcessorConfigDTO. The frequency with which to schedule the processor. The format of the value will depend on the value of schedulingStrategy.

**Returns** The scheduling\_period of this ProcessorConfigDTO.

**Return type** str

**scheduling\_strategy**

Gets the `scheduling_strategy` of this `ProcessorConfigDTO`. Indicates whether the processor should be scheduled to run in event or timer driven mode.

**Returns** The `scheduling_strategy` of this `ProcessorConfigDTO`.

**Return type** `str`

**sensitive\_dynamic\_property\_names**

Gets the `sensitive_dynamic_property_names` of this `ProcessorConfigDTO`. Set of sensitive dynamic property names

**Returns** The `sensitive_dynamic_property_names` of this `ProcessorConfigDTO`.

**Return type** `list[str]`

**swagger\_types** = {'annotation\_data': 'str', 'auto\_terminated\_relationships': 'list[str]

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**yield\_duration**

Gets the `yield_duration` of this `ProcessorConfigDTO`. The amount of time that must elapse before this processor is scheduled again after yielding.

**Returns** The `yield_duration` of this `ProcessorConfigDTO`.

**Return type** `str`

**nipyapi.nifi.models.processor\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_dto.ProcessorDTO (id=None, versioned_component_id=None, parent_group_id=None, position=None, name=None, type=None, bundle=None, state=None, style=None, relationships=None, description=None, supports_parallel_processing=None, supports_event_driven=None, supports_batching=None, supports_sensitive_dynamic_properties=None, persists_state=None, restricted=None, deprecated=None, execution_node_restricted=None, multiple_versions_available=None, input_requirement=None, config=None, validation_errors=None, validation_status=None, extension_missing=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorDTO - a model defined in Swagger

```
attribute_map = {'bundle': 'bundle', 'config': 'config', 'deprecated': 'deprecated'}
```

#### **bundle**

Gets the bundle of this ProcessorDTO. The details of the artifact that bundled this processor type.

**Returns** The bundle of this ProcessorDTO.

**Return type** *BundleDTO*

#### **config**

Gets the config of this ProcessorDTO. The configuration details for the processor. These details will be included in a response if the verbose flag is included in a request.

**Returns** The config of this ProcessorDTO.

**Return type** *ProcessorConfigDTO*

#### **deprecated**

Gets the deprecated of this ProcessorDTO. Whether the processor has been deprecated.

**Returns** The deprecated of this ProcessorDTO.

**Return type** bool

#### **description**

Gets the description of this ProcessorDTO. The description of the processor.

**Returns** The description of this ProcessorDTO.

**Return type** str

**execution\_node\_restricted**

Gets the execution\_node\_restricted of this ProcessorDTO. Indicates if the execution node of a processor is restricted to run only on the primary node

**Returns** The execution\_node\_restricted of this ProcessorDTO.

**Return type** bool

**extension\_missing**

Gets the extension\_missing of this ProcessorDTO. Whether the underlying extension is missing.

**Returns** The extension\_missing of this ProcessorDTO.

**Return type** bool

**id**

Gets the id of this ProcessorDTO. The id of the component.

**Returns** The id of this ProcessorDTO.

**Return type** str

**input\_requirement**

Gets the input\_requirement of this ProcessorDTO. The input requirement for this processor.

**Returns** The input\_requirement of this ProcessorDTO.

**Return type** str

**multiple\_versions\_available**

Gets the multiple\_versions\_available of this ProcessorDTO. Whether the processor has multiple versions available.

**Returns** The multiple\_versions\_available of this ProcessorDTO.

**Return type** bool

**name**

Gets the name of this ProcessorDTO. The name of the processor.

**Returns** The name of this ProcessorDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ProcessorDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ProcessorDTO.

**Return type** str

**persists\_state**

Gets the persists\_state of this ProcessorDTO. Whether the processor persists state.

**Returns** The persists\_state of this ProcessorDTO.

**Return type** bool

**position**

Gets the position of this ProcessorDTO. The position of this component in the UI if applicable.

**Returns** The position of this ProcessorDTO.

**Return type** *PositionDTO*

**relationships**

Gets the relationships of this ProcessorDTO. The available relationships that the processor currently supports.

**Returns** The relationships of this ProcessorDTO.

**Return type** list[*RelationshipDTO*]

**restricted**

Gets the restricted of this ProcessorDTO. Whether the processor requires elevated privileges.

**Returns** The restricted of this ProcessorDTO.

**Return type** bool

**state**

Gets the state of this ProcessorDTO. The state of the processor

**Returns** The state of this ProcessorDTO.

**Return type** str

**style**

Gets the style of this ProcessorDTO. Styles for the processor (background-color : #eee).

**Returns** The style of this ProcessorDTO.

**Return type** dict(str, str)

**supports\_batching**

Gets the supports\_batching of this ProcessorDTO. Whether the processor supports batching. This makes the run duration settings available.

**Returns** The supports\_batching of this ProcessorDTO.

**Return type** bool

**supports\_event\_driven**

Gets the supports\_event\_driven of this ProcessorDTO. Whether the processor supports event driven scheduling.

**Returns** The supports\_event\_driven of this ProcessorDTO.

**Return type** bool

**supports\_parallel\_processing**

Gets the supports\_parallel\_processing of this ProcessorDTO. Whether the processor supports parallel processing.

**Returns** The supports\_parallel\_processing of this ProcessorDTO.

**Return type** bool

**supports\_sensitive\_dynamic\_properties**

Gets the supports\_sensitive\_dynamic\_properties of this ProcessorDTO. Whether the processor supports sensitive dynamic properties.

**Returns** The supports\_sensitive\_dynamic\_properties of this ProcessorDTO.

**Return type** bool

**swagger\_types** = {'bundle': 'BundleDTO', 'config': 'ProcessorConfigDTO', 'deprecated'

**to\_dict** ()

Returns the model properties as a dict



**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ProcessorDTO. The type of the processor.

**Returns** The type of this ProcessorDTO.

**Return type** str

**validation\_errors**

Gets the validation\_errors of this ProcessorDTO. The validation errors for the processor. These validation errors represent the problems with the processor that must be resolved before it can be started.

**Returns** The validation\_errors of this ProcessorDTO.

**Return type** list[str]

**validation\_status**

Gets the validation\_status of this ProcessorDTO. Indicates whether the Processor is valid, invalid, or still in the process of validating (i.e., it is unknown whether or not the Processor is valid)

**Returns** The validation\_status of this ProcessorDTO.

**Return type** str

**versioned\_component\_id**

Gets the versioned\_component\_id of this ProcessorDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ProcessorDTO.

**Return type** str

## nipyapi.nifi.models.processor\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_entity.ProcessorEntity (revision=None,
                                                         id=None, uri=None,
                                                         position=None, per-
                                                         missions=None, bul-
                                                         letins=None, discon-
                                                         nected_node_acknowledged=None,
                                                         component=None, in-
                                                         put_requirement=None,
                                                         status=None, oper-
                                                         ate_permissions=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this ProcessorEntity. The bulletins for this component.

**Returns** The bulletins of this ProcessorEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this ProcessorEntity.

**Returns** The component of this ProcessorEntity.

**Return type** *ProcessorDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ProcessorEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ProcessorEntity.

**Return type** bool

**id**

Gets the id of this ProcessorEntity. The id of the component.

**Returns** The id of this ProcessorEntity.

**Return type** str

**input\_requirement**

Gets the input\_requirement of this ProcessorEntity. The input requirement for this processor.

**Returns** The input\_requirement of this ProcessorEntity.

**Return type** str

**operate\_permissions**

Gets the operate\_permissions of this ProcessorEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this ProcessorEntity.

**Return type** *PermissionsDTO*

**permissions**

Gets the permissions of this ProcessorEntity. The permissions for this component.

**Returns** The permissions of this ProcessorEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this ProcessorEntity. The position of this component in the UI if applicable.

**Returns** The position of this ProcessorEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this ProcessorEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ProcessorEntity.

**Return type** *RevisionDTO*

**status**

Gets the status of this ProcessorEntity.

**Returns** The status of this ProcessorEntity.

**Return type** *ProcessorStatusDTO*

```
swagger_types = {'bulletins': 'list[BulletinEntity]', 'component': 'ProcessorDTO', 'uri': 'uri'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this ProcessorEntity. The URI for futures requests to the component.

**Returns** The uri of this ProcessorEntity.

**Return type** str

### nipyapi.nifi.models.processor\_status\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_status_dto.ProcessorStatusDTO (group_id=None,
                                                                    id=None,
                                                                    name=None,
                                                                    type=None,
                                                                    run_status=None,
                                                                    stats_last_refreshed=None,
                                                                    aggregate_snapshot=None,
                                                                    node_snapshots=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorStatusDTO - a model defined in Swagger

#### aggregate\_snapshot

Gets the aggregate\_snapshot of this ProcessorStatusDTO. A status snapshot that represents the aggregate stats of all nodes in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this represents the stats of the single instance.

**Returns** The aggregate\_snapshot of this ProcessorStatusDTO.

**Return type** *ProcessorStatusSnapshotDTO*

```
attribute_map = {'aggregate_snapshot': 'aggregateSnapshot', 'group_id': 'groupId', 'id': 'id', 'name': 'name', 'run_status': 'runStatus', 'stats_last_refreshed': 'statsLastRefreshed', 'type': 'type', 'uri': 'uri'}
```

**group\_id**

Gets the group\_id of this ProcessorStatusDTO. The unique ID of the process group that the Processor belongs to

**Returns** The group\_id of this ProcessorStatusDTO.

**Return type** str

**id**

Gets the id of this ProcessorStatusDTO. The unique ID of the Processor

**Returns** The id of this ProcessorStatusDTO.

**Return type** str

**name**

Gets the name of this ProcessorStatusDTO. The name of the Processor

**Returns** The name of this ProcessorStatusDTO.

**Return type** str

**node\_snapshots**

Gets the node\_snapshots of this ProcessorStatusDTO. A status snapshot for each node in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this may be null.

**Returns** The node\_snapshots of this ProcessorStatusDTO.

**Return type** list[*NodeProcessorStatusSnapshotDTO*]

**run\_status**

Gets the run\_status of this ProcessorStatusDTO. The run status of the Processor

**Returns** The run\_status of this ProcessorStatusDTO.

**Return type** str

**stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this ProcessorStatusDTO. The timestamp of when the stats were last refreshed

**Returns** The stats\_last\_refreshed of this ProcessorStatusDTO.

**Return type** str

**swagger\_types** = {'aggregate\_snapshot': 'ProcessorStatusSnapshotDTO', 'group\_id': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ProcessorStatusDTO. The type of the Processor

**Returns** The type of this ProcessorStatusDTO.

**Return type** str

## nipyapi.nifi.models.processor\_status\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_status_entity.ProcessorStatusEntity(processor_status=None,  
                                                                    can_read=None)  
    Bases: object  
  
    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
    ProcessorStatusEntity - a model defined in Swagger  
  
    attribute_map = {'can_read': 'canRead', 'processor_status': 'processorStatus'}  
  
    can_read  
        Gets the can_read of this ProcessorStatusEntity. Indicates whether the user can read a given resource.  
        Returns The can_read of this ProcessorStatusEntity.  
        Return type bool  
  
    processor_status  
        Gets the processor_status of this ProcessorStatusEntity.  
        Returns The processor_status of this ProcessorStatusEntity.  
        Return type ProcessorStatusDTO  
  
    swagger_types = {'can_read': 'bool', 'processor_status': 'ProcessorStatusDTO'}  
  
    to_dict()  
        Returns the model properties as a dict  
  
    to_str()  
        Returns the string representation of the model
```

### nipyapi.nifi.models.processor\_status\_snapshot\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO (id=None,
group_id=None,
name=None,
type=None,
run_status=None,
execution_node=None,
bytes_read=None,
bytes_written=None,
read=None,
written=None,
flow_files_read=None,
bytes_in=None,
bytes_out=None,
tasks_running=None,
tasks_duration=None,
active_threads=None,
terminated_threads=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorStatusSnapshotDTO - a model defined in Swagger

#### **active\_thread\_count**

Gets the active\_thread\_count of this ProcessorStatusSnapshotDTO. The number of threads currently executing in the processor.

**Returns** The active\_thread\_count of this ProcessorStatusSnapshotDTO.

**Return type** int

**attribute\_map** = {'active\_thread\_count': 'activeThreadCount', 'bytes\_in': 'bytesIn',

#### **bytes\_in**

Gets the bytes\_in of this ProcessorStatusSnapshotDTO. The size of the FlowFiles that have been accepted in the last 5 minutes

**Returns** The bytes\_in of this ProcessorStatusSnapshotDTO.

**Return type** int

#### **bytes\_out**

Gets the bytes\_out of this ProcessorStatusSnapshotDTO. The size of the FlowFiles transferred to a Connection in the last 5 minutes

**Returns** The bytes\_out of this ProcessorStatusSnapshotDTO.

**Return type** int

**bytes\_read**

Gets the bytes\_read of this ProcessorStatusSnapshotDTO. The number of bytes read by this Processor in the last 5 minutes

**Returns** The bytes\_read of this ProcessorStatusSnapshotDTO.

**Return type** int

**bytes\_written**

Gets the bytes\_written of this ProcessorStatusSnapshotDTO. The number of bytes written by this Processor in the last 5 minutes

**Returns** The bytes\_written of this ProcessorStatusSnapshotDTO.

**Return type** int

**execution\_node**

Gets the execution\_node of this ProcessorStatusSnapshotDTO. Indicates the node where the process will execute.

**Returns** The execution\_node of this ProcessorStatusSnapshotDTO.

**Return type** str

**flow\_files\_in**

Gets the flow\_files\_in of this ProcessorStatusSnapshotDTO. The number of FlowFiles that have been accepted in the last 5 minutes

**Returns** The flow\_files\_in of this ProcessorStatusSnapshotDTO.

**Return type** int

**flow\_files\_out**

Gets the flow\_files\_out of this ProcessorStatusSnapshotDTO. The number of FlowFiles transferred to a Connection in the last 5 minutes

**Returns** The flow\_files\_out of this ProcessorStatusSnapshotDTO.

**Return type** int

**group\_id**

Gets the group\_id of this ProcessorStatusSnapshotDTO. The id of the parent process group to which the processor belongs.

**Returns** The group\_id of this ProcessorStatusSnapshotDTO.

**Return type** str

**id**

Gets the id of this ProcessorStatusSnapshotDTO. The id of the processor.

**Returns** The id of this ProcessorStatusSnapshotDTO.

**Return type** str

**input**

Gets the input of this ProcessorStatusSnapshotDTO. The count/size of flowfiles that have been accepted in the last 5 minutes.

**Returns** The input of this ProcessorStatusSnapshotDTO.

**Return type** str

**name**

Gets the name of this ProcessorStatusSnapshotDTO. The name of the processor.

**Returns** The name of this ProcessorStatusSnapshotDTO.

**Return type** str

**output**

Gets the output of this ProcessorStatusSnapshotDTO. The count/size of flowfiles that have been processed in the last 5 minutes.

**Returns** The output of this ProcessorStatusSnapshotDTO.

**Return type** str

**read**

Gets the read of this ProcessorStatusSnapshotDTO. The number of bytes read in the last 5 minutes.

**Returns** The read of this ProcessorStatusSnapshotDTO.

**Return type** str

**run\_status**

Gets the run\_status of this ProcessorStatusSnapshotDTO. The state of the processor.

**Returns** The run\_status of this ProcessorStatusSnapshotDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'bytes\_in': 'int', 'bytes\_out': 'int'

**task\_count**

Gets the task\_count of this ProcessorStatusSnapshotDTO. The number of times this Processor has run in the last 5 minutes

**Returns** The task\_count of this ProcessorStatusSnapshotDTO.

**Return type** int

**tasks**

Gets the tasks of this ProcessorStatusSnapshotDTO. The total number of task this connectable has completed over the last 5 minutes.

**Returns** The tasks of this ProcessorStatusSnapshotDTO.

**Return type** str

**tasks\_duration**

Gets the tasks\_duration of this ProcessorStatusSnapshotDTO. The total duration of all tasks for this connectable over the last 5 minutes.

**Returns** The tasks\_duration of this ProcessorStatusSnapshotDTO.

**Return type** str

**tasks\_duration\_nanos**

Gets the tasks\_duration\_nanos of this ProcessorStatusSnapshotDTO. The number of nanoseconds that this Processor has spent running in the last 5 minutes

**Returns** The tasks\_duration\_nanos of this ProcessorStatusSnapshotDTO.

**Return type** int

**terminated\_thread\_count**

Gets the terminated\_thread\_count of this ProcessorStatusSnapshotDTO. The number of threads currently terminated for the processor.



**Returns** The `terminated_thread_count` of this `ProcessorStatusSnapshotDTO`.

**Return type** `int`

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this `ProcessorStatusSnapshotDTO`. The type of the processor.

**Returns** The type of this `ProcessorStatusSnapshotDTO`.

**Return type** `str`

**written**

Gets the written of this `ProcessorStatusSnapshotDTO`. The number of bytes written in the last 5 minutes.

**Returns** The written of this `ProcessorStatusSnapshotDTO`.

**Return type** `str`

## nipyapi.nifi.models.processor\_status\_snapshot\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity` (*id*

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

`ProcessorStatusSnapshotEntity` - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'id': 'id', 'processor\_status\_snapshot': 'p

**can\_read**

Gets the `can_read` of this `ProcessorStatusSnapshotEntity`. Indicates whether the user can read a given resource.

**Returns** The `can_read` of this `ProcessorStatusSnapshotEntity`.

**Return type** `bool`

**id**

Gets the `id` of this `ProcessorStatusSnapshotEntity`. The `id` of the processor.

**Returns** The `id` of this `ProcessorStatusSnapshotEntity`.

**Return type** `str`

**processor\_status\_snapshot**

Gets the processor\_status\_snapshot of this ProcessorStatusSnapshotEntity.

**Returns** The processor\_status\_snapshot of this ProcessorStatusSnapshotEntity.

**Return type** *ProcessorStatusSnapshotDTO*

**swagger\_types** = {'can\_read': 'bool', 'id': 'str', 'processor\_status\_snapshot': 'Pro

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.processor\_types\_entity module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.processor\_types\_entity.**ProcessorTypesEntity** (*processor\_types=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorTypesEntity - a model defined in Swagger

**attribute\_map** = {'processor\_types': 'processorTypes'}

**processor\_types**

Gets the processor\_types of this ProcessorTypesEntity.

**Returns** The processor\_types of this ProcessorTypesEntity.

**Return type** list[*DocumentedTypeDTO*]

**swagger\_types** = {'processor\_types': 'list[DocumentedTypeDTO] '}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.processors\_entity module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.processors_entity.ProcessorsEntity (processors=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProcessorsEntity - a model defined in Swagger

```
attribute_map = {'processors':  'processors'}
```

```
processors
```

Gets the processors of this ProcessorsEntity.

**Returns** The processors of this ProcessorsEntity.

**Return type** list[*ProcessorEntity*]

```
swagger_types = {'processors':  'list[ProcessorEntity]'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

### nipyapi.nifi.models.property\_descriptor\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO (name=None,
                                                                           dis-
                                                                           play_name=None,
                                                                           de-
                                                                           scrip-
                                                                           tion=None,
                                                                           de-
                                                                           fault_value=None,
                                                                           al-
                                                                           low-
                                                                           able_values=None,
                                                                           re-
                                                                           quired=None,
                                                                           sen-
                                                                           si-
                                                                           tive=None,
                                                                           dy-
                                                                           namic=None,
                                                                           sup-
                                                                           ports_el=None,
                                                                           ex-
                                                                           pres-
                                                                           sion_language_scope=None,
                                                                           iden-
                                                                           ti-
                                                                           fies_controller_service=None,
                                                                           iden-
                                                                           ti-
                                                                           fies_controller_service_bundle_id=None,
                                                                           de-
                                                                           pen-
                                                                           den-
                                                                           cies=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PropertyDescriptorDTO - a model defined in Swagger

#### **allowable\_values**

Gets the allowable\_values of this PropertyDescriptorDTO. Allowable values for the property. If empty then the allowed values are not constrained.

**Returns** The allowable\_values of this PropertyDescriptorDTO.

**Return type** list[[AllowableValueEntity](#)]

**attribute\_map** = {'allowable\_values': 'allowableValues', 'default\_value': 'defaultVal

#### **default\_value**

Gets the default\_value of this PropertyDescriptorDTO. The default value for the property.

**Returns** The default\_value of this PropertyDescriptorDTO.

**Return type** str

#### **dependencies**

Gets the dependencies of this PropertyDescriptorDTO. A list of dependencies that must be met in order

for this Property to be relevant. If any of these dependencies is not met, the property described by this Property Descriptor is not relevant.

**Returns** The dependencies of this PropertyDescriptorDTO.

**Return type** list[PropertyDependencyDTO]

**description**

Gets the description of this PropertyDescriptorDTO. The description for the property. Used to relay additional details to a user or provide a mechanism of documenting intent.

**Returns** The description of this PropertyDescriptorDTO.

**Return type** str

**display\_name**

Gets the display\_name of this PropertyDescriptorDTO. The human readable name for the property.

**Returns** The display\_name of this PropertyDescriptorDTO.

**Return type** str

**dynamic**

Gets the dynamic of this PropertyDescriptorDTO. Whether the property is dynamic (user-defined).

**Returns** The dynamic of this PropertyDescriptorDTO.

**Return type** bool

**expression\_language\_scope**

Gets the expression\_language\_scope of this PropertyDescriptorDTO. Scope of the Expression Language evaluation for the property.

**Returns** The expression\_language\_scope of this PropertyDescriptorDTO.

**Return type** str

**identifies\_controller\_service**

Gets the identifies\_controller\_service of this PropertyDescriptorDTO. If the property identifies a controller service this returns the fully qualified type.

**Returns** The identifies\_controller\_service of this PropertyDescriptorDTO.

**Return type** str

**identifies\_controller\_service\_bundle**

Gets the identifies\_controller\_service\_bundle of this PropertyDescriptorDTO. If the property identifies a controller service this returns the bundle of the type, null otherwise.

**Returns** The identifies\_controller\_service\_bundle of this PropertyDescriptorDTO.

**Return type** *BundleDTO*

**name**

Gets the name of this PropertyDescriptorDTO. The name for the property.

**Returns** The name of this PropertyDescriptorDTO.

**Return type** str

**required**

Gets the required of this PropertyDescriptorDTO. Whether the property is required.

**Returns** The required of this PropertyDescriptorDTO.

**Return type** bool

**sensitive**

Gets the sensitive of this PropertyDescriptorDTO. Whether the property is sensitive and protected whenever stored or represented.

**Returns** The sensitive of this PropertyDescriptorDTO.

**Return type** bool

**supports\_el**

Gets the supports\_el of this PropertyDescriptorDTO. Whether the property supports expression language.

**Returns** The supports\_el of this PropertyDescriptorDTO.

**Return type** bool

**swagger\_types** = {'allowable\_values': 'list[AllowableValueEntity]', 'default\_value':

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.property\_descriptor\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.property\_descriptor\_entity.**PropertyDescriptorEntity**(*property\_descriptor*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PropertyDescriptorEntity - a model defined in Swagger

**attribute\_map** = {'property\_descriptor': 'propertyDescriptor'}

**property\_descriptor**

Gets the property\_descriptor of this PropertyDescriptorEntity.

**Returns** The property\_descriptor of this PropertyDescriptorEntity.

**Return type** *PropertyDescriptorDTO*

**swagger\_types** = {'property\_descriptor': 'PropertyDescriptor'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.property\_history\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.property\_history\_dto.**PropertyHistoryDTO** (*previous\_values=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

PropertyHistoryDTO - a model defined in Swagger

**attribute\_map** = {'previous\_values': 'previousValues'}

**previous\_values**

Gets the previous\_values of this PropertyHistoryDTO. Previous values for a given property.

**Returns** The previous\_values of this PropertyHistoryDTO.

**Return type** list[*PreviousValueDTO*]

**swagger\_types** = {'previous\_values': 'list[PreviousValueDTO]'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_dto.**ProvenanceDTO** (*id=None, uri=None, submission\_time=None, expiration=None, percent\_completed=None, finished=None, request=None, results=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceDTO - a model defined in Swagger

**attribute\_map** = {'expiration': 'expiration', 'finished': 'finished', 'id': 'id', 'p

**expiration**

Gets the expiration of this ProvenanceDTO. The timestamp when the query will expire.

**Returns** The expiration of this ProvenanceDTO.

**Return type** str

**finished**

Gets the finished of this ProvenanceDTO. Whether the query has finished.

**Returns** The finished of this ProvenanceDTO.

**Return type** bool

**id**

Gets the id of this ProvenanceDTO. The id of the provenance query.

**Returns** The id of this ProvenanceDTO.

**Return type** str

**percent\_completed**

Gets the percent\_completed of this ProvenanceDTO. The current percent complete.

**Returns** The percent\_completed of this ProvenanceDTO.

**Return type** int

**request**

Gets the request of this ProvenanceDTO. The provenance request.

**Returns** The request of this ProvenanceDTO.

**Return type** *ProvenanceRequestDTO*

**results**

Gets the results of this ProvenanceDTO. The provenance results.

**Returns** The results of this ProvenanceDTO.

**Return type** *ProvenanceResultsDTO*

**submission\_time**

Gets the submission\_time of this ProvenanceDTO. The timestamp when the query was submitted.

**Returns** The submission\_time of this ProvenanceDTO.

**Return type** str

**swagger\_types** = {'expiration': 'str', 'finished': 'bool', 'id': 'str', 'percent\_com

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this ProvenanceDTO. The URI for this query. Used for obtaining/deleting the request at a later time

**Returns** The uri of this ProvenanceDTO.

**Return type** str

## nipyapi.nifi.models.provenance\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.



OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_entity.**ProvenanceEntity** (*provenance=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceEntity - a model defined in Swagger

**attribute\_map** = {'provenance': 'provenance'}

**provenance**

Gets the provenance of this ProvenanceEntity.

**Returns** The provenance of this ProvenanceEntity.

**Return type** *ProvenanceDTO*

**swagger\_types** = {'provenance': 'ProvenanceDTO'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

### nipyapi.nifi.models.provenance\_event\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO (id=None,  
                                                                event_id=None,  
                                                                event_time=None,  
                                                                event_duration=None,  
                                                                lin-  
                                                                event_duration=None,  
                                                                event_type=None,  
                                                                flow_file_uuid=None,  
                                                                file_size=None,  
                                                                file_size_bytes=None,  
                                                                clus-  
                                                                ter_node_id=None,  
                                                                clus-  
                                                                ter_node_address=None,  
                                                                group_id=None,  
                                                                compo-  
                                                                nent_id=None,  
                                                                compo-  
                                                                nent_type=None,  
                                                                compo-  
                                                                nent_name=None,  
                                                                source_system_flow_file_id=None,  
                                                                alter-  
                                                                nate_identifier_uri=None,  
                                                                at-  
                                                                tributes=None,  
                                                                par-  
                                                                ent_uuids=None,  
                                                                child_uuids=None,  
                                                                tran-  
                                                                sit_uri=None,  
                                                                relation-  
                                                                ship=None,  
                                                                de-  
                                                                tails=None,  
                                                                con-  
                                                                tent_equal=None,  
                                                                in-  
                                                                put_content_available=None,  
                                                                in-  
                                                                put_content_claim_section=None,  
                                                                in-  
                                                                put_content_claim_container=None,  
                                                                in-  
                                                                put_content_claim_identifier=None,  
                                                                in-  
                                                                put_content_claim_offset=None,  
                                                                in-  
                                                                put_content_claim_file_size=None,  
                                                                in-  
                                                                put_content_claim_file_size_bytes=None,  
                                                                out-  
                                                                put_content_available=None,  
                                                                out-  
                                                                put_content_claim_section=None,  
                                                                out-  
                                                                put_content_claim_container=None,  
                                                                out-  
                                                                put_content_claim_identifier=None,  
                                                                out-
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceEventDTO - a model defined in Swagger

**alternate\_identifier\_uri**

Gets the alternate\_identifier\_uri of this ProvenanceEventDTO. The alternate identifier uri for the fileflow for the event.

**Returns** The alternate\_identifier\_uri of this ProvenanceEventDTO.

**Return type** `str`

**attribute\_map** = {'alternate\_identifier\_uri': 'alternateIdentifierUri', 'attributes':

**attributes**

Gets the attributes of this ProvenanceEventDTO. The attributes of the flowfile for the event.

**Returns** The attributes of this ProvenanceEventDTO.

**Return type** `list[AttributeDTO]`

**child\_uuids**

Gets the child\_uuids of this ProvenanceEventDTO. The child uuids for the event.

**Returns** The child\_uuids of this ProvenanceEventDTO.

**Return type** `list[str]`

**cluster\_node\_address**

Gets the cluster\_node\_address of this ProvenanceEventDTO. The label for the node where the event originated.

**Returns** The cluster\_node\_address of this ProvenanceEventDTO.

**Return type** `str`

**cluster\_node\_id**

Gets the cluster\_node\_id of this ProvenanceEventDTO. The identifier for the node where the event originated.

**Returns** The cluster\_node\_id of this ProvenanceEventDTO.

**Return type** `str`

**component\_id**

Gets the component\_id of this ProvenanceEventDTO. The id of the component that generated the event.

**Returns** The component\_id of this ProvenanceEventDTO.

**Return type** `str`

**component\_name**

Gets the component\_name of this ProvenanceEventDTO. The name of the component that generated the event.

**Returns** The component\_name of this ProvenanceEventDTO.

**Return type** `str`

**component\_type**

Gets the component\_type of this ProvenanceEventDTO. The type of the component that generated the event.

**Returns** The component\_type of this ProvenanceEventDTO.

**Return type** str

**content\_equal**

Gets the content\_equal of this ProvenanceEventDTO. Whether the input and output content claim is the same.

**Returns** The content\_equal of this ProvenanceEventDTO.

**Return type** bool

**details**

Gets the details of this ProvenanceEventDTO. The event details.

**Returns** The details of this ProvenanceEventDTO.

**Return type** str

**event\_duration**

Gets the event\_duration of this ProvenanceEventDTO. The event duration in milliseconds.

**Returns** The event\_duration of this ProvenanceEventDTO.

**Return type** int

**event\_id**

Gets the event\_id of this ProvenanceEventDTO. The event id. This is a one up number thats unique per node.

**Returns** The event\_id of this ProvenanceEventDTO.

**Return type** int

**event\_time**

Gets the event\_time of this ProvenanceEventDTO. The timestamp of the event.

**Returns** The event\_time of this ProvenanceEventDTO.

**Return type** str

**event\_type**

Gets the event\_type of this ProvenanceEventDTO. The type of the event.

**Returns** The event\_type of this ProvenanceEventDTO.

**Return type** str

**file\_size**

Gets the file\_size of this ProvenanceEventDTO. The size of the flowfile for the event.

**Returns** The file\_size of this ProvenanceEventDTO.

**Return type** str

**file\_size\_bytes**

Gets the file\_size\_bytes of this ProvenanceEventDTO. The size of the flowfile in bytes for the event.

**Returns** The file\_size\_bytes of this ProvenanceEventDTO.

**Return type** int

**flow\_file\_uuid**

Gets the flow\_file\_uuid of this ProvenanceEventDTO. The uuid of the flowfile for the event.

**Returns** The flow\_file\_uuid of this ProvenanceEventDTO.

**Return type** str

**group\_id**

Gets the group\_id of this ProvenanceEventDTO. The id of the group that the component resides in. If the component is no longer in the flow, the group id will not be set.

**Returns** The group\_id of this ProvenanceEventDTO.

**Return type** str

**id**

Gets the id of this ProvenanceEventDTO. The event uuid.

**Returns** The id of this ProvenanceEventDTO.

**Return type** str

**input\_content\_available**

Gets the input\_content\_available of this ProvenanceEventDTO. Whether the input content is still available.

**Returns** The input\_content\_available of this ProvenanceEventDTO.

**Return type** bool

**input\_content\_claim\_container**

Gets the input\_content\_claim\_container of this ProvenanceEventDTO. The container in which the input content claim lives.

**Returns** The input\_content\_claim\_container of this ProvenanceEventDTO.

**Return type** str

**input\_content\_claim\_file\_size**

Gets the input\_content\_claim\_file\_size of this ProvenanceEventDTO. The file size of the input content claim formatted.

**Returns** The input\_content\_claim\_file\_size of this ProvenanceEventDTO.

**Return type** str

**input\_content\_claim\_file\_size\_bytes**

Gets the input\_content\_claim\_file\_size\_bytes of this ProvenanceEventDTO. The file size of the input content claim in bytes.

**Returns** The input\_content\_claim\_file\_size\_bytes of this ProvenanceEventDTO.

**Return type** int

**input\_content\_claim\_identifier**

Gets the input\_content\_claim\_identifier of this ProvenanceEventDTO. The identifier of the input content claim.

**Returns** The input\_content\_claim\_identifier of this ProvenanceEventDTO.

**Return type** str

**input\_content\_claim\_offset**

Gets the input\_content\_claim\_offset of this ProvenanceEventDTO. The offset into the input content claim where the flowfiles content begins.

**Returns** The input\_content\_claim\_offset of this ProvenanceEventDTO.

**Return type** int

**input\_content\_claim\_section**

Gets the input\_content\_claim\_section of this ProvenanceEventDTO. The section in which the input content claim lives.

**Returns** The input\_content\_claim\_section of this ProvenanceEventDTO.

**Return type** str

**lineage\_duration**

Gets the lineage\_duration of this ProvenanceEventDTO. The duration since the lineage began, in milliseconds.

**Returns** The lineage\_duration of this ProvenanceEventDTO.

**Return type** int

**output\_content\_available**

Gets the output\_content\_available of this ProvenanceEventDTO. Whether the output content is still available.

**Returns** The output\_content\_available of this ProvenanceEventDTO.

**Return type** bool

**output\_content\_claim\_container**

Gets the output\_content\_claim\_container of this ProvenanceEventDTO. The container in which the output content claim lives.

**Returns** The output\_content\_claim\_container of this ProvenanceEventDTO.

**Return type** str

**output\_content\_claim\_file\_size**

Gets the output\_content\_claim\_file\_size of this ProvenanceEventDTO. The file size of the output content claim formatted.

**Returns** The output\_content\_claim\_file\_size of this ProvenanceEventDTO.

**Return type** str

**output\_content\_claim\_file\_size\_bytes**

Gets the output\_content\_claim\_file\_size\_bytes of this ProvenanceEventDTO. The file size of the output content claim in bytes.

**Returns** The output\_content\_claim\_file\_size\_bytes of this ProvenanceEventDTO.

**Return type** int

**output\_content\_claim\_identifier**

Gets the output\_content\_claim\_identifier of this ProvenanceEventDTO. The identifier of the output content claim.

**Returns** The output\_content\_claim\_identifier of this ProvenanceEventDTO.

**Return type** str

**output\_content\_claim\_offset**

Gets the output\_content\_claim\_offset of this ProvenanceEventDTO. The offset into the output content claim where the flowfiles content begins.

**Returns** The output\_content\_claim\_offset of this ProvenanceEventDTO.

**Return type** int

**output\_content\_claim\_section**

Gets the output\_content\_claim\_section of this ProvenanceEventDTO. The section in which the output content claim lives.

**Returns** The output\_content\_claim\_section of this ProvenanceEventDTO.

**Return type** str

**parent\_uuids**

Gets the parent\_uuids of this ProvenanceEventDTO. The parent uuids for the event.

**Returns** The parent\_uuids of this ProvenanceEventDTO.

**Return type** list[str]

**relationship**

Gets the relationship of this ProvenanceEventDTO. The relationship to which the flowfile was routed if the event is of type ROUTE.

**Returns** The relationship of this ProvenanceEventDTO.

**Return type** str

**replay\_available**

Gets the replay\_available of this ProvenanceEventDTO. Whether or not replay is available.

**Returns** The replay\_available of this ProvenanceEventDTO.

**Return type** bool

**replay\_explanation**

Gets the replay\_explanation of this ProvenanceEventDTO. Explanation as to why replay is unavailable.

**Returns** The replay\_explanation of this ProvenanceEventDTO.

**Return type** str

**source\_connection\_identifier**

Gets the source\_connection\_identifier of this ProvenanceEventDTO. The identifier of the queue/connection from which the flowfile was pulled to generate this event. May be null if the queue/connection is unknown or the flowfile was generated from this event.

**Returns** The source\_connection\_identifier of this ProvenanceEventDTO.

**Return type** str

**source\_system\_flow\_file\_id**

Gets the source\_system\_flow\_file\_id of this ProvenanceEventDTO. The source system flowfile id.

**Returns** The source\_system\_flow\_file\_id of this ProvenanceEventDTO.

**Return type** str

**swagger\_types** = {'alternate\_identifier\_uri': 'str', 'attributes': 'list[AttributeDTO

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transit\_uri**

Gets the transit\_uri of this ProvenanceEventDTO. The source/destination system uri if the event was a RECEIVE/SEND.

**Returns** The transit\_uri of this ProvenanceEventDTO.

**Return type** str

## nipyapi.nifi.models.provenance\_event\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_event\_entity.**ProvenanceEventEntity** (*provenance\_event=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceEventEntity - a model defined in Swagger

**attribute\_map** = {'provenance\_event': 'provenanceEvent'}

**provenance\_event**

Gets the provenance\_event of this ProvenanceEventEntity.

**Returns** The provenance\_event of this ProvenanceEventEntity.

**Return type** *ProvenanceEventDTO*

**swagger\_types** = {'provenance\_event': 'ProvenanceEventDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_link\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_link\_dto.**ProvenanceLinkDTO** (*source\_id=None, tar-  
get\_id=None, flow\_file\_uuid=None, times-  
tamp=None, millis=None*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceLinkDTO - a model defined in Swagger

**attribute\_map** = {'flow\_file\_uuid': 'flowFileUuid', 'millis': 'millis', 'source\_id':



**flow\_file\_uuid**

Gets the flow\_file\_uuid of this ProvenanceLinkDTO. The flowfile uuid that traversed the link.

**Returns** The flow\_file\_uuid of this ProvenanceLinkDTO.

**Return type** str

**millis**

Gets the millis of this ProvenanceLinkDTO. The timestamp of this link in milliseconds.

**Returns** The millis of this ProvenanceLinkDTO.

**Return type** int

**source\_id**

Gets the source\_id of this ProvenanceLinkDTO. The source node id of the link.

**Returns** The source\_id of this ProvenanceLinkDTO.

**Return type** str

**swagger\_types** = {'flow\_file\_uuid': 'str', 'millis': 'int', 'source\_id': 'str', 'target\_id': 'str', 'timestamp': 'str', 'to\_dict': 'dict', 'to\_str': 'str'}

**target\_id**

Gets the target\_id of this ProvenanceLinkDTO. The target node id of the link.

**Returns** The target\_id of this ProvenanceLinkDTO.

**Return type** str

**timestamp**

Gets the timestamp of this ProvenanceLinkDTO. The timestamp of the link (based on the destination).

**Returns** The timestamp of this ProvenanceLinkDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_node\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.provenance_node_dto.ProvenanceNodeDTO (id=None,  
flow_file_uuid=None,  
parent_uuids=None,  
child_uuids=None,  
cluster_node_identifier=None,  
type=None,  
event_type=None,  
millis=None,  
timestamp=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceNodeDTO - a model defined in Swagger

```
attribute_map = {'child_uuids': 'childUuids', 'cluster_node_identifier': 'clusterNodeIdentifier'}
```

#### **child\_uuids**

Gets the child\_uuids of this ProvenanceNodeDTO. The uuid of the childrent flowfiles of the provenance event.

**Returns** The child\_uuids of this ProvenanceNodeDTO.

**Return type** list[str]

#### **cluster\_node\_identifier**

Gets the cluster\_node\_identifier of this ProvenanceNodeDTO. The identifier of the node that this event/flowfile originated from.

**Returns** The cluster\_node\_identifier of this ProvenanceNodeDTO.

**Return type** str

#### **event\_type**

Gets the event\_type of this ProvenanceNodeDTO. If the type is EVENT, this is the type of event.

**Returns** The event\_type of this ProvenanceNodeDTO.

**Return type** str

#### **flow\_file\_uuid**

Gets the flow\_file\_uuid of this ProvenanceNodeDTO. The uuid of the flowfile associated with the provenance event.

**Returns** The flow\_file\_uuid of this ProvenanceNodeDTO.

**Return type** str

#### **id**

Gets the id of this ProvenanceNodeDTO. The id of the node.

**Returns** The id of this ProvenanceNodeDTO.

**Return type** str

#### **millis**

Gets the millis of this ProvenanceNodeDTO. The timestamp of the node in milliseconds.

**Returns** The millis of this ProvenanceNodeDTO.

**Return type** int

**parent\_uuids**

Gets the parent\_uuids of this ProvenanceNodeDTO. The uuid of the parent flowfiles of the provenance event.

**Returns** The parent\_uuids of this ProvenanceNodeDTO.

**Return type** list[str]

**swagger\_types** = {'child\_uuids': 'list[str]', 'cluster\_node\_identifier': 'str', 'event'

**timestamp**

Gets the timestamp of this ProvenanceNodeDTO. The timestamp of the node formatted.

**Returns** The timestamp of this ProvenanceNodeDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ProvenanceNodeDTO. The type of the node.

**Returns** The type of this ProvenanceNodeDTO.

**Return type** str

**nipyapi.nifi.models.provenance\_options\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_options\_dto.**ProvenanceOptionsDTO** (*searchable\_fields=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceOptionsDTO - a model defined in Swagger

**attribute\_map** = {'searchable\_fields': 'searchableFields'}

**searchable\_fields**

Gets the searchable\_fields of this ProvenanceOptionsDTO. The available searchable field for the NiFi.

**Returns** The searchable\_fields of this ProvenanceOptionsDTO.

**Return type** list[ProvenanceSearchableFieldDTO]

**swagger\_types** = {'searchable\_fields': 'list[ProvenanceSearchableFieldDTO]'}  
'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_options\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.provenance\_options\_entity.**ProvenanceOptionsEntity** (*provenance\_options=*  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceOptionsEntity - a model defined in Swagger

**attribute\_map** = {'provenance\_options': 'provenanceOptions'}

**provenance\_options**

Gets the provenance\_options of this ProvenanceOptionsEntity.

**Returns** The provenance\_options of this ProvenanceOptionsEntity.

**Return type** *ProvenanceOptionsDTO*

**swagger\_types** = {'provenance\_options': 'ProvenanceOptionsDTO'}

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_request\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDTO(search_terms=None,
                                                                    cluster_node_id=None,
                                                                    start_date=None,
                                                                    end_date=None,
                                                                    minimal_file_size=None,
                                                                    maximum_file_size=None,
                                                                    max_results=None,
                                                                    summarize=None,
                                                                    incremental_results=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceRequestDTO - a model defined in Swagger

```
attribute_map = {'cluster_node_id': 'clusterNodeId', 'end_date': 'endDate', 'incremental_results': 'incrementalResults'}
```

**cluster\_node\_id**

Gets the cluster\_node\_id of this ProvenanceRequestDTO. The id of the node in the cluster where this provenance originated.

**Returns** The cluster\_node\_id of this ProvenanceRequestDTO.

**Return type** str

**end\_date**

Gets the end\_date of this ProvenanceRequestDTO. The latest event time to include in the query.

**Returns** The end\_date of this ProvenanceRequestDTO.

**Return type** str

**incremental\_results**

Gets the incremental\_results of this ProvenanceRequestDTO. Whether or not incremental results are returned. If false, provenance events are only returned once the query completes. This property is true by default.

**Returns** The incremental\_results of this ProvenanceRequestDTO.

**Return type** bool

**max\_results**

Gets the max\_results of this ProvenanceRequestDTO. The maximum number of results to include.

**Returns** The max\_results of this ProvenanceRequestDTO.

**Return type** int

**maximum\_file\_size**

Gets the maximum\_file\_size of this ProvenanceRequestDTO. The maximum file size to include in the query.

**Returns** The maximum\_file\_size of this ProvenanceRequestDTO.

**Return type** str

**minimum\_file\_size**

Gets the `minimum_file_size` of this `ProvenanceRequestDTO`. The minimum file size to include in the query.

**Returns** The `minimum_file_size` of this `ProvenanceRequestDTO`.

**Return type** `str`

**search\_terms**

Gets the `search_terms` of this `ProvenanceRequestDTO`. The search terms used to perform the search.

**Returns** The `search_terms` of this `ProvenanceRequestDTO`.

**Return type** `dict(str, ProvenanceSearchValueDTO)`

**start\_date**

Gets the `start_date` of this `ProvenanceRequestDTO`. The earliest event time to include in the query.

**Returns** The `start_date` of this `ProvenanceRequestDTO`.

**Return type** `str`

**summarize**

Gets the `summarize` of this `ProvenanceRequestDTO`. Whether or not to summarize provenance events returned. This property is `false` by default.

**Returns** The `summarize` of this `ProvenanceRequestDTO`.

**Return type** `bool`

**swagger\_types** = {'cluster\_node\_id': 'str', 'end\_date': 'str', 'incremental\_results':

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.provenance\_results\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.provenance_results_dto.ProvenanceResultsDTO(provenance_events=None,
                                                                    to-
                                                                    tal=None,
                                                                    to-
                                                                    tal_count=None,
                                                                    gen-
                                                                    er-
                                                                    ated=None,
                                                                    old-
                                                                    est_event=None,
                                                                    time_offset=None,
                                                                    er-
                                                                    rors=None)
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceResultsDTO - a model defined in Swagger

**attribute\_map** = {'errors': 'errors', 'generated': 'generated', 'oldest\_event': 'oldest\_event', 'total': 'total', 'total\_count': 'total\_count', 'time\_offset': 'time\_offset'}

#### **errors**

Gets the errors of this ProvenanceResultsDTO. Any errors that occurred while performing the provenance request.

**Returns** The errors of this ProvenanceResultsDTO.

**Return type** `list[str]`

#### **generated**

Gets the generated of this ProvenanceResultsDTO. Then the search was performed.

**Returns** The generated of this ProvenanceResultsDTO.

**Return type** `str`

#### **oldest\_event**

Gets the oldest\_event of this ProvenanceResultsDTO. The oldest event available in the provenance repository.

**Returns** The oldest\_event of this ProvenanceResultsDTO.

**Return type** `str`

#### **provenance\_events**

Gets the provenance\_events of this ProvenanceResultsDTO. The provenance events that matched the search criteria.

**Returns** The provenance\_events of this ProvenanceResultsDTO.

**Return type** `list[ProvenanceEventDTO]`

**swagger\_types** = {'errors': 'list[str]', 'generated': 'str', 'oldest\_event': 'str', 'total': 'int', 'total\_count': 'int', 'time\_offset': 'int'}

#### **time\_offset**

Gets the time\_offset of this ProvenanceResultsDTO. The time offset of the server that's used for event time.

**Returns** The time\_offset of this ProvenanceResultsDTO.

**Return type** `int`

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **total**

Gets the total of this ProvenanceResultsDTO. The total number of results formatted.

**Returns** The total of this ProvenanceResultsDTO.

**Return type** `str`

#### **total\_count**

Gets the total\_count of this ProvenanceResultsDTO. The total number of results.

**Returns** The total\_count of this ProvenanceResultsDTO.

**Return type** int

## nipyapi.nifi.models.provenance\_searchable\_field\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.provenance_searchable_field_dto.ProvenanceSearchableFieldDTO (id=N,
                                                                                       field=,
                                                                                       la=,
                                                                                       bel=,
                                                                                       type=)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ProvenanceSearchableFieldDTO - a model defined in Swagger

```
attribute_map = {'field': 'field', 'id': 'id', 'label': 'label', 'type': 'type'}
```

**field**

Gets the field of this ProvenanceSearchableFieldDTO. The searchable field.

**Returns** The field of this ProvenanceSearchableFieldDTO.

**Return type** str

**id**

Gets the id of this ProvenanceSearchableFieldDTO. The id of the searchable field.

**Returns** The id of this ProvenanceSearchableFieldDTO.

**Return type** str

**label**

Gets the label of this ProvenanceSearchableFieldDTO. The label for the searchable field.

**Returns** The label of this ProvenanceSearchableFieldDTO.

**Return type** str

```
swagger_types = {'field': 'str', 'id': 'str', 'label': 'str', 'type': 'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ProvenanceSearchableFieldDTO. The type of the searchable field.

**Returns** The type of this ProvenanceSearchableFieldDTO.

**Return type** str



## nipyapi.nifi.models.queue\_size\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.queue_size_dto.QueueSizeDTO (byte_count=None,      ob-
                                                    ject_count=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

QueueSizeDTO - a model defined in Swagger

```
attribute_map = {'byte_count':  'byteCount', 'object_count':  'objectCount'}
```

#### byte\_count

Gets the byte\_count of this QueueSizeDTO. The size of objects in a queue.

**Returns** The byte\_count of this QueueSizeDTO.

**Return type** int

#### object\_count

Gets the object\_count of this QueueSizeDTO. The count of objects in a queue.

**Returns** The object\_count of this QueueSizeDTO.

**Return type** int

```
swagger_types = {'byte_count':  'int', 'object_count':  'int'}
```

#### to\_dict()

Returns the model properties as a dict

#### to\_str()

Returns the string representation of the model

## nipyapi.nifi.models.registry\_client\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.registry_client_entity.RegistryClientEntity (revision=None,
                                                                    id=None,
                                                                    uri=None,
                                                                    position=None,
                                                                    permissions=None,
                                                                    bulletins=None,
                                                                    disconnected_node_acknowledged=None,
                                                                    component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RegistryClientEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'component': 'component', 'disconnected\_node\_acknowledged': 'disconnected\_node\_acknowledged', 'id': 'id', 'permissions': 'permissions', 'position': 'position', 'uri': 'uri'}

**bulletins**

Gets the bulletins of this RegistryClientEntity. The bulletins for this component.

**Returns** The bulletins of this RegistryClientEntity.

**Return type** list[[BulletinEntity](#)]

**component**

Gets the component of this RegistryClientEntity.

**Returns** The component of this RegistryClientEntity.

**Return type** [RegistryDTO](#)

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this RegistryClientEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this RegistryClientEntity.

**Return type** bool

**id**

Gets the id of this RegistryClientEntity. The id of the component.

**Returns** The id of this RegistryClientEntity.

**Return type** str

**permissions**

Gets the permissions of this RegistryClientEntity. The permissions for this component.

**Returns** The permissions of this RegistryClientEntity.

**Return type** [PermissionsDTO](#)

**position**

Gets the position of this RegistryClientEntity. The position of this component in the UI if applicable.

**Returns** The position of this RegistryClientEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this RegistryClientEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this RegistryClientEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'RegistryDTO', 'd

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this RegistryClientEntity. The URI for futures requests to the component.

**Returns** The uri of this RegistryClientEntity.

**Return type** str

## nipyapi.nifi.models.registry\_clients\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.registry\_clients\_entity.**RegistryClientsEntity** (registries=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RegistryClientsEntity - a model defined in Swagger

**attribute\_map** = {'registries': 'registries'}

**registries**

Gets the registries of this RegistryClientsEntity.

**Returns** The registries of this RegistryClientsEntity.

**Return type** list[*RegistryClientEntity*]

**swagger\_types** = {'registries': 'list[RegistryClientEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.registry\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.registry_dto.RegistryDTO (id=None, name=None, description=None, uri=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RegistryDTO - a model defined in Swagger

```
attribute_map = {'description': 'description', 'id': 'id', 'name': 'name', 'uri':
```

```
description
```

Gets the description of this RegistryDTO. The registry description

**Returns** The description of this RegistryDTO.

**Return type** str

```
id
```

Gets the id of this RegistryDTO. The registry identifier

**Returns** The id of this RegistryDTO.

**Return type** str

```
name
```

Gets the name of this RegistryDTO. The registry name

**Returns** The name of this RegistryDTO.

**Return type** str

```
swagger_types = {'description': 'str', 'id': 'str', 'name': 'str', 'uri': 'str'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

```
uri
```

Gets the uri of this RegistryDTO. The registry URI

**Returns** The uri of this RegistryDTO.

**Return type** str

## nipyapi.nifi.models.relationship\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.relationship_dto.RelationshipDTO (name=None, de-  
scription=None,  
auto_terminate=None,  
retry=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RelationshipDTO - a model defined in Swagger

```
attribute_map = {'auto_terminate': 'autoTerminate', 'description': 'description', 'name': 'name', 'retry': 'retry'}
```

**auto\_terminate**

Gets the auto\_terminate of this RelationshipDTO. Whether or not flowfiles sent to this relationship should auto terminate.

**Returns** The auto\_terminate of this RelationshipDTO.

**Return type** bool

**description**

Gets the description of this RelationshipDTO. The relationship description.

**Returns** The description of this RelationshipDTO.

**Return type** str

**name**

Gets the name of this RelationshipDTO. The relationship name.

**Returns** The name of this RelationshipDTO.

**Return type** str

**retry**

Gets the retry of this RelationshipDTO. Whether or not flowfiles sent to this relationship should retry.

**Returns** The retry of this RelationshipDTO.

**Return type** bool

```
swagger_types = {'auto_terminate': 'bool', 'description': 'str', 'name': 'str', 'retry': 'bool'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

### nipyapi.nifi.models.remote\_process\_group\_contents\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.remote_process_group_contents_dto.RemoteProcessGroupContentsDTO (i
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupContentsDTO - a model defined in Swagger

```
attribute_map = {'input_ports': 'inputPorts', 'output_ports': 'outputPorts'}
```

**input\_ports**

Gets the input\_ports of this RemoteProcessGroupContentsDTO. The input ports to which data can be sent.

**Returns** The input\_ports of this RemoteProcessGroupContentsDTO.

**Return type** list[RemoteProcessGroupPortDTO]

**output\_ports**

Gets the output\_ports of this RemoteProcessGroupContentsDTO. The output ports from which data can be retrieved.

**Returns** The output\_ports of this RemoteProcessGroupContentsDTO.

**Return type** list[RemoteProcessGroupPortDTO]

```
swagger_types = {'input_ports': 'list[RemoteProcessGroupPortDTO]', 'output_ports': 'list[RemoteProcessGroupPortDTO]'
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.remote\_process\_group\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO (id=None,
                                                                    ver-
                                                                    sioned_component_id=None,
                                                                    par-
                                                                    ent_group_id=None,
                                                                    po-
                                                                    si-
                                                                    tion=None,
                                                                    tar-
                                                                    get_uri=None,
                                                                    tar-
                                                                    get_uris=None,
                                                                    tar-
                                                                    get_secure=None,
                                                                    name=None,
                                                                    com-
                                                                    ments=None,
                                                                    com-
                                                                    mu-
                                                                    ni-
                                                                    ca-
                                                                    tions_timeout=None,
                                                                    yield_duration=None,
                                                                    trans-
                                                                    port_protocol=None,
                                                                    lo-
                                                                    cal_network_interface=None,
                                                                    proxy_host=None,
                                                                    proxy_port=None,
                                                                    proxy_user=None,
                                                                    proxy_password=None,
                                                                    au-
                                                                    tho-
                                                                    riza-
                                                                    tion_issues=None,
                                                                    val-
                                                                    i-
                                                                    da-
                                                                    tion_errors=None,
                                                                    trans-
                                                                    mit-
                                                                    ting=None,
                                                                    in-
                                                                    put_port_count=None,
                                                                    out-
                                                                    put_port_count=None,
                                                                    ac-
                                                                    tive_remote_input_port_c
                                                                    in-
                                                                    ac-
                                                                    tive_remote_input_port_c
                                                                    ac-
                                                                    tive_remote_output_port_
                                                                    in-
                                                                    ac-
                                                                    tive_remote_output_port_
                                                                    flow_id=freshed=None,
                                                                    con-
                                                                    tents=None)

```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupDTO - a model defined in Swagger

**active\_remote\_input\_port\_count**

Gets the active\_remote\_input\_port\_count of this RemoteProcessGroupDTO. The number of active remote input ports.

**Returns** The active\_remote\_input\_port\_count of this RemoteProcessGroupDTO.

**Return type** `int`

**active\_remote\_output\_port\_count**

Gets the active\_remote\_output\_port\_count of this RemoteProcessGroupDTO. The number of active remote output ports.

**Returns** The active\_remote\_output\_port\_count of this RemoteProcessGroupDTO.

**Return type** `int`

**attribute\_map** = {'active\_remote\_input\_port\_count': 'activeRemoteInputPortCount', 'act.

**authorization\_issues**

Gets the authorization\_issues of this RemoteProcessGroupDTO. Any remote authorization issues for the remote process group.

**Returns** The authorization\_issues of this RemoteProcessGroupDTO.

**Return type** `list[str]`

**comments**

Gets the comments of this RemoteProcessGroupDTO. The comments for the remote process group.

**Returns** The comments of this RemoteProcessGroupDTO.

**Return type** `str`

**communications\_timeout**

Gets the communications\_timeout of this RemoteProcessGroupDTO. The time period used for the timeout when communicating with the target.

**Returns** The communications\_timeout of this RemoteProcessGroupDTO.

**Return type** `str`

**contents**

Gets the contents of this RemoteProcessGroupDTO. The contents of the remote process group. Will contain available input/output ports.

**Returns** The contents of this RemoteProcessGroupDTO.

**Return type** *RemoteProcessGroupContentsDTO*

**flow\_refreshed**

Gets the flow\_refreshed of this RemoteProcessGroupDTO. The timestamp when this remote process group was last refreshed.

**Returns** The flow\_refreshed of this RemoteProcessGroupDTO.

**Return type** `str`

**id**

Gets the id of this RemoteProcessGroupDTO. The id of the component.

**Returns** The id of this RemoteProcessGroupDTO.



**Return type** str

**inactive\_remote\_input\_port\_count**

Gets the inactive\_remote\_input\_port\_count of this RemoteProcessGroupDTO. The number of inactive remote input ports.

**Returns** The inactive\_remote\_input\_port\_count of this RemoteProcessGroupDTO.

**Return type** int

**inactive\_remote\_output\_port\_count**

Gets the inactive\_remote\_output\_port\_count of this RemoteProcessGroupDTO. The number of inactive remote output ports.

**Returns** The inactive\_remote\_output\_port\_count of this RemoteProcessGroupDTO.

**Return type** int

**input\_port\_count**

Gets the input\_port\_count of this RemoteProcessGroupDTO. The number of remote input ports currently available on the target.

**Returns** The input\_port\_count of this RemoteProcessGroupDTO.

**Return type** int

**local\_network\_interface**

Gets the local\_network\_interface of this RemoteProcessGroupDTO. The local network interface to send/receive data. If not specified, any local address is used. If clustered, all nodes must have an interface with this identifier.

**Returns** The local\_network\_interface of this RemoteProcessGroupDTO.

**Return type** str

**name**

Gets the name of this RemoteProcessGroupDTO. The name of the remote process group.

**Returns** The name of this RemoteProcessGroupDTO.

**Return type** str

**output\_port\_count**

Gets the output\_port\_count of this RemoteProcessGroupDTO. The number of remote output ports currently available on the target.

**Returns** The output\_port\_count of this RemoteProcessGroupDTO.

**Return type** int

**parent\_group\_id**

Gets the parent\_group\_id of this RemoteProcessGroupDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this RemoteProcessGroupDTO.

**Return type** str

**position**

Gets the position of this RemoteProcessGroupDTO. The position of this component in the UI if applicable.

**Returns** The position of this RemoteProcessGroupDTO.

**Return type** *PositionDTO*

**proxy\_host**

Gets the proxy\_host of this RemoteProcessGroupDTO.

**Returns** The proxy\_host of this RemoteProcessGroupDTO.

**Return type** str

**proxy\_password**

Gets the proxy\_password of this RemoteProcessGroupDTO.

**Returns** The proxy\_password of this RemoteProcessGroupDTO.

**Return type** str

**proxy\_port**

Gets the proxy\_port of this RemoteProcessGroupDTO.

**Returns** The proxy\_port of this RemoteProcessGroupDTO.

**Return type** int

**proxy\_user**

Gets the proxy\_user of this RemoteProcessGroupDTO.

**Returns** The proxy\_user of this RemoteProcessGroupDTO.

**Return type** str

**swagger\_types** = {'active\_remote\_input\_port\_count': 'int', 'active\_remote\_output\_port\_

**target\_secure**

Gets the target\_secure of this RemoteProcessGroupDTO. Whether the target is running securely.

**Returns** The target\_secure of this RemoteProcessGroupDTO.

**Return type** bool

**target\_uri**

Gets the target\_uri of this RemoteProcessGroupDTO. The target URI of the remote process group. If target uri is not set, but uris are set, then returns the first url in the urls. If neither target uri nor uris are set, then returns null.

**Returns** The target\_uri of this RemoteProcessGroupDTO.

**Return type** str

**target\_uris**

Gets the target\_uris of this RemoteProcessGroupDTO. The target URI of the remote process group. If target uris is not set but target uri is set, then returns a collection containing the single target uri. If neither target uris nor uris are set, then returns null.

**Returns** The target\_uris of this RemoteProcessGroupDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmitting**

Gets the transmitting of this RemoteProcessGroupDTO. Whether the remote process group is actively transmitting.

**Returns** The transmitting of this RemoteProcessGroupDTO.

**Return type** bool

**transport\_protocol**

Gets the transport\_protocol of this RemoteProcessGroupDTO.

**Returns** The transport\_protocol of this RemoteProcessGroupDTO.

**Return type** str

**validation\_errors**

Gets the validation\_errors of this RemoteProcessGroupDTO. The validation errors for the remote process group. These validation errors represent the problems with the remote process group that must be resolved before it can transmit.

**Returns** The validation\_errors of this RemoteProcessGroupDTO.

**Return type** list[str]

**versioned\_component\_id**

Gets the versioned\_component\_id of this RemoteProcessGroupDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this RemoteProcessGroupDTO.

**Return type** str

**yield\_duration**

Gets the yield\_duration of this RemoteProcessGroupDTO. When yielding, this amount of time must elapse before the remote process group is scheduled again.

**Returns** The yield\_duration of this RemoteProcessGroupDTO.

**Return type** str

## nipyapi.nifi.models.remote\_process\_group\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity (revision=None,
id=None,
uri=None,
po-
si-
tion=None,
per-
mis-
sions=None,
bul-
letins=None,
dis-
con-
nected_node_ack-
com-
po-
nent=None,
sta-
tus=None,
in-
put_port_count=
out-
put_port_count=
op-
er-
ate_permissions=
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupEntity - a model defined in Swagger

```
attribute_map = {'bulletins':  'bulletins', 'component':  'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this RemoteProcessGroupEntity. The bulletins for this component.

**Returns** The bulletins of this RemoteProcessGroupEntity.

**Return type** list[[BulletinEntity](#)]

**component**

Gets the component of this RemoteProcessGroupEntity.

**Returns** The component of this RemoteProcessGroupEntity.

**Return type** [RemoteProcessGroupDTO](#)

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this RemoteProcessGroupEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this RemoteProcessGroupEntity.

**Return type** bool

**id**

Gets the id of this RemoteProcessGroupEntity. The id of the component.

**Returns** The id of this RemoteProcessGroupEntity.

**Return type** str

**input\_port\_count**

Gets the `input_port_count` of this `RemoteProcessGroupEntity`. The number of remote input ports currently available on the target.

**Returns** The `input_port_count` of this `RemoteProcessGroupEntity`.

**Return type** `int`

**operate\_permissions**

Gets the `operate_permissions` of this `RemoteProcessGroupEntity`. The permissions for this component operations.

**Returns** The `operate_permissions` of this `RemoteProcessGroupEntity`.

**Return type** *PermissionsDTO*

**output\_port\_count**

Gets the `output_port_count` of this `RemoteProcessGroupEntity`. The number of remote output ports currently available on the target.

**Returns** The `output_port_count` of this `RemoteProcessGroupEntity`.

**Return type** `int`

**permissions**

Gets the permissions of this `RemoteProcessGroupEntity`. The permissions for this component.

**Returns** The permissions of this `RemoteProcessGroupEntity`.

**Return type** *PermissionsDTO*

**position**

Gets the position of this `RemoteProcessGroupEntity`. The position of this component in the UI if applicable.

**Returns** The position of this `RemoteProcessGroupEntity`.

**Return type** *PositionDTO*

**revision**

Gets the revision of this `RemoteProcessGroupEntity`. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this `RemoteProcessGroupEntity`.

**Return type** *RevisionDTO*

**status**

Gets the status of this `RemoteProcessGroupEntity`. The status of the remote process group.

**Returns** The status of this `RemoteProcessGroupEntity`.

**Return type** *RemoteProcessGroupStatusDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'RemoteProcessGroupEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this `RemoteProcessGroupEntity`. The URI for futures requests to the component.

**Returns** The uri of this `RemoteProcessGroupEntity`.

**Return type** str

## nipyapi.nifi.models.remote\_process\_group\_port\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.remote_process_group_port_dto.RemoteProcessGroupPortDTO (id=None,
                                          tar-
                                          get_id=None,
                                          ver-
                                          sioned_comp
                                          group_id=None,
                                          name=None,
                                          com-
                                          ments=None,
                                          con-
                                          cur-
                                          rently_sched
                                          trans-
                                          mit-
                                          ting=None,
                                          use_compres
                                          ex-
                                          ists=None,
                                          tar-
                                          get_running,
                                          con-
                                          nected=None,
                                          batch_setting,
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupPortDTO - a model defined in Swagger

```
attribute_map = {'batch_settings': 'batchSettings', 'comments': 'comments', 'concurr
```

#### **batch\_settings**

Gets the batch\_settings of this RemoteProcessGroupPortDTO. The batch settings for data transmission.

**Returns** The batch\_settings of this RemoteProcessGroupPortDTO.

**Return type** *BatchSettingsDTO*

#### **comments**

Gets the comments of this RemoteProcessGroupPortDTO. The comments as configured on the target port.

**Returns** The comments of this RemoteProcessGroupPortDTO.

**Return type** str

**concurrently\_schedulable\_task\_count**

Gets the `concurrently_schedulable_task_count` of this `RemoteProcessGroupPortDTO`. The number of task that may transmit flowfiles to the target port concurrently.

**Returns** The `concurrently_schedulable_task_count` of this `RemoteProcessGroupPortDTO`.

**Return type** `int`

**connected**

Gets the `connected` of this `RemoteProcessGroupPortDTO`. Whether the port has either an incoming or outgoing connection.

**Returns** The `connected` of this `RemoteProcessGroupPortDTO`.

**Return type** `bool`

**exists**

Gets the `exists` of this `RemoteProcessGroupPortDTO`. Whether the target port exists.

**Returns** The `exists` of this `RemoteProcessGroupPortDTO`.

**Return type** `bool`

**group\_id**

Gets the `group_id` of this `RemoteProcessGroupPortDTO`. The id of the remote process group that the port resides in.

**Returns** The `group_id` of this `RemoteProcessGroupPortDTO`.

**Return type** `str`

**id**

Gets the `id` of this `RemoteProcessGroupPortDTO`. The id of the port.

**Returns** The `id` of this `RemoteProcessGroupPortDTO`.

**Return type** `str`

**name**

Gets the `name` of this `RemoteProcessGroupPortDTO`. The name of the target port.

**Returns** The `name` of this `RemoteProcessGroupPortDTO`.

**Return type** `str`

**swagger\_types** = {'batch\_settings': 'BatchSettingsDTO', 'comments': 'str', 'concurrent

**target\_id**

Gets the `target_id` of this `RemoteProcessGroupPortDTO`. The id of the target port.

**Returns** The `target_id` of this `RemoteProcessGroupPortDTO`.

**Return type** `str`

**target\_running**

Gets the `target_running` of this `RemoteProcessGroupPortDTO`. Whether the target port is running.

**Returns** The `target_running` of this `RemoteProcessGroupPortDTO`.

**Return type** `bool`

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmitting**

Gets the transmitting of this RemoteProcessGroupPortDTO. Whether the remote port is configured for transmission.

**Returns** The transmitting of this RemoteProcessGroupPortDTO.

**Return type** bool

**use\_compression**

Gets the use\_compression of this RemoteProcessGroupPortDTO. Whether the flowfiles are compressed when sent to the target port.

**Returns** The use\_compression of this RemoteProcessGroupPortDTO.

**Return type** bool

**versioned\_component\_id**

Gets the versioned\_component\_id of this RemoteProcessGroupPortDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this RemoteProcessGroupPortDTO.

**Return type** str

**nipyapi.nifi.models.remote\_process\_group\_port\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity (rev

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupPortEntity - a model defined in Swagger



```
attribute_map = {'bulletins': 'bulletins', 'disconnected_node_acknowledged': 'discon
```

#### **bulletins**

Gets the bulletins of this RemoteProcessGroupPortEntity. The bulletins for this component.

**Returns** The bulletins of this RemoteProcessGroupPortEntity.

**Return type** list[*BulletinEntity*]

#### **disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this RemoteProcessGroupPortEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this RemoteProcessGroupPortEntity.

**Return type** bool

#### **id**

Gets the id of this RemoteProcessGroupPortEntity. The id of the component.

**Returns** The id of this RemoteProcessGroupPortEntity.

**Return type** str

#### **operate\_permissions**

Gets the operate\_permissions of this RemoteProcessGroupPortEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this RemoteProcessGroupPortEntity.

**Return type** *PermissionsDTO*

#### **permissions**

Gets the permissions of this RemoteProcessGroupPortEntity. The permissions for this component.

**Returns** The permissions of this RemoteProcessGroupPortEntity.

**Return type** *PermissionsDTO*

#### **position**

Gets the position of this RemoteProcessGroupPortEntity. The position of this component in the UI if applicable.

**Returns** The position of this RemoteProcessGroupPortEntity.

**Return type** *PositionDTO*

#### **remote\_process\_group\_port**

Gets the remote\_process\_group\_port of this RemoteProcessGroupPortEntity.

**Returns** The remote\_process\_group\_port of this RemoteProcessGroupPortEntity.

**Return type** *RemoteProcessGroupPortDTO*

#### **revision**

Gets the revision of this RemoteProcessGroupPortEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this RemoteProcessGroupPortEntity.

**Return type** *RevisionDTO*

```
swagger_types = {'bulletins': 'list[BulletinEntity]', 'disconnected_node_acknowledged
```

#### **to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this RemoteProcessGroupPortEntity. The URI for futures requests to the component.

**Returns** The uri of this RemoteProcessGroupPortEntity.

**Return type** str

## nipyapi.nifi.models.remote\_process\_group\_status\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.remote_process_group_status_dto.RemoteProcessGroupStatusDTO (group_id=None, name=None, target_uri=None, transport=None, mission_status=None, stats=None, value=None, id=None, data=None, transaction=None, aggregate=None, gate=None, node=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupStatusDTO - a model defined in Swagger

#### **aggregate\_snapshot**

Gets the aggregate\_snapshot of this RemoteProcessGroupStatusDTO. A status snapshot that represents the aggregate stats of all nodes in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this represents the stats of the single instance.

**Returns** The aggregate\_snapshot of this RemoteProcessGroupStatusDTO.

**Return type** RemoteProcessGroupStatusSnapshotDTO

**attribute\_map** = {'aggregate\_snapshot': 'aggregateSnapshot', 'group\_id': 'groupId', 'id': 'id', 'mission\_status': 'missionStatus', 'stats': 'stats', 'target\_uri': 'targetUri', 'transport': 'transport', 'value': 'value', 'transaction': 'transaction', 'aggregate': 'aggregate', 'gate': 'gate', 'node': 'node'}

#### **group\_id**

Gets the group\_id of this RemoteProcessGroupStatusDTO. The unique ID of the process group that the Processor belongs to

**Returns** The group\_id of this RemoteProcessGroupStatusDTO.

**Return type** str

**id**

Gets the id of this RemoteProcessGroupStatusDTO. The unique ID of the Processor

**Returns** The id of this RemoteProcessGroupStatusDTO.

**Return type** str

**name**

Gets the name of this RemoteProcessGroupStatusDTO. The name of the remote process group.

**Returns** The name of this RemoteProcessGroupStatusDTO.

**Return type** str

**node\_snapshots**

Gets the node\_snapshots of this RemoteProcessGroupStatusDTO. A status snapshot for each node in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this may be null.

**Returns** The node\_snapshots of this RemoteProcessGroupStatusDTO.

**Return type** list[*NodeRemoteProcessGroupStatusSnapshotDTO*]

**stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this RemoteProcessGroupStatusDTO. The time the status for the process group was last refreshed.

**Returns** The stats\_last\_refreshed of this RemoteProcessGroupStatusDTO.

**Return type** str

**swagger\_types** = {'aggregate\_snapshot': 'RemoteProcessGroupStatusSnapshotDTO', 'group\_

**target\_uri**

Gets the target\_uri of this RemoteProcessGroupStatusDTO. The URI of the target system.

**Returns** The target\_uri of this RemoteProcessGroupStatusDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmission\_status**

Gets the transmission\_status of this RemoteProcessGroupStatusDTO. The transmission status of the remote process group.

**Returns** The transmission\_status of this RemoteProcessGroupStatusDTO.

**Return type** str

**validation\_status**

Gets the validation\_status of this RemoteProcessGroupStatusDTO. Indicates whether the component is valid, invalid, or still in the process of validating (i.e., it is unknown whether or not the component is valid)

**Returns** The validation\_status of this RemoteProcessGroupStatusDTO.

**Return type** str

## nipyapi.nifi.models.remote\_process\_group\_status\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupStatusEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'remote\_process\_group\_status': 'remoteProcessGroupStatus'}

**can\_read**

Gets the can\_read of this RemoteProcessGroupStatusEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this RemoteProcessGroupStatusEntity.

**Return type** bool

**remote\_process\_group\_status**

Gets the remote\_process\_group\_status of this RemoteProcessGroupStatusEntity.

**Returns** The remote\_process\_group\_status of this RemoteProcessGroupStatusEntity.

**Return type** RemoteProcessGroupStatusDTO

**swagger\_types** = {'can\_read': 'bool', 'remote\_process\_group\_status': 'RemoteProcessGroupStatusDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatus
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupStatusSnapshotDTO - a model defined in Swagger

**active\_thread\_count**

Gets the active\_thread\_count of this RemoteProcessGroupStatusSnapshotDTO. The number of active threads for the remote process group.

**Returns** The active\_thread\_count of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** int

```
attribute_map = {'active_thread_count': 'activeThreadCount', 'bytes_received': 'byte
```

**bytes\_received**

Gets the bytes\_received of this RemoteProcessGroupStatusSnapshotDTO. The size of the FlowFiles received from the remote process group in the last 5 minutes.

**Returns** The bytes\_received of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** int

**bytes\_sent**

Gets the bytes\_sent of this RemoteProcessGroupStatusSnapshotDTO. The size of the FlowFiles sent to the remote process group in the last 5 minutes.

**Returns** The bytes\_sent of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_received**

Gets the flow\_files\_received of this RemoteProcessGroupStatusSnapshotDTO. The number of FlowFiles received from the remote process group in the last 5 minutes.

**Returns** The flow\_files\_received of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** int

**flow\_files\_sent**

Gets the flow\_files\_sent of this RemoteProcessGroupStatusSnapshotDTO. The number of FlowFiles sent to the remote process group in the last 5 minutes.

**Returns** The flow\_files\_sent of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** int

**group\_id**

Gets the group\_id of this RemoteProcessGroupStatusSnapshotDTO. The id of the parent process group the remote process group resides in.

**Returns** The group\_id of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**id**

Gets the id of this RemoteProcessGroupStatusSnapshotDTO. The id of the remote process group.

**Returns** The id of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**name**

Gets the name of this RemoteProcessGroupStatusSnapshotDTO. The name of the remote process group.

**Returns** The name of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**received**

Gets the received of this RemoteProcessGroupStatusSnapshotDTO. The count/size of the flowfiles received from the remote process group in the last 5 minutes.

**Returns** The received of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**sent**

Gets the sent of this RemoteProcessGroupStatusSnapshotDTO. The count/size of the flowfiles sent to the remote process group in the last 5 minutes.

**Returns** The sent of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**swagger\_types** = {'active\_thread\_count': 'int', 'bytes\_received': 'int', 'bytes\_sent'

**target\_uri**

Gets the target\_uri of this RemoteProcessGroupStatusSnapshotDTO. The URI of the target system.

**Returns** The target\_uri of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transmission\_status**

Gets the transmission\_status of this RemoteProcessGroupStatusSnapshotDTO. The transmission status of the remote process group.

**Returns** The transmission\_status of this RemoteProcessGroupStatusSnapshotDTO.

**Return type** str

## nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.nifi.models.remote_process_group_status_snapshot_entity.RemoteProcessGroupStatusSnapshotEntity`

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupStatusSnapshotEntity - a model defined in Swagger

**attribute\_map** = {'can\_read': 'canRead', 'id': 'id', 'remote\_process\_group\_status\_snapshot': 'remoteProcessGroupStatusSnapshot'}

#### **can\_read**

Gets the can\_read of this RemoteProcessGroupStatusSnapshotEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this RemoteProcessGroupStatusSnapshotEntity.

**Return type** `bool`

#### **id**

Gets the id of this RemoteProcessGroupStatusSnapshotEntity. The id of the remote process group.

**Returns** The id of this RemoteProcessGroupStatusSnapshotEntity.

**Return type** `str`

#### **remote\_process\_group\_status\_snapshot**

Gets the remote\_process\_group\_status\_snapshot of this RemoteProcessGroupStatusSnapshotEntity.

**Returns** The remote\_process\_group\_status\_snapshot of this RemoteProcessGroupStatusSnapshotEntity.

**Return type** `RemoteProcessGroupStatusSnapshotDTO`

**swagger\_types** = {'can\_read': 'bool', 'id': 'str', 'remote\_process\_group\_status\_snapshot': 'RemoteProcessGroupStatusSnapshotDTO'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.remote\_process\_groups\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity (remote\_proce

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RemoteProcessGroupsEntity - a model defined in Swagger

**attribute\_map** = {'remote\_process\_groups': 'remoteProcessGroups'}

**remote\_process\_groups**

Gets the remote\_process\_groups of this RemoteProcessGroupsEntity.

**Returns** The remote\_process\_groups of this RemoteProcessGroupsEntity.

**Return type** list[RemoteProcessGroupEntity]

**swagger\_types** = {'remote\_process\_groups': 'list[RemoteProcessGroupEntity]'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

### nipyapi.nifi.models.reporting\_task\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>



```

class nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO (id=None, ver-
sioned_component_id=None, par-
ent_group_id=None,
position=None,
name=None,
type=None,
bundle=None,
state=None, com-
ments=None, per-
sists_state=None,
restricted=None,
depre-
cated=None,
multi-
ple_versions_available=None,
sup-
ports_sensitive_dynamic_properties=None,
schedul-
ing_period=None,
schedul-
ing_strategy=None,
de-
fault_scheduling_period=None,
properties=None,
descrip-
tors=None, sensi-
tive_dynamic_property_names=None,
cus-
tom_ui_url=None,
annota-
tion_data=None,
valida-
tion_errors=None,
valida-
tion_status=None,
ac-
tive_thread_count=None,
exten-
sion_missing=None)

```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ReportingTaskDTO - a model defined in Swagger

#### **active\_thread\_count**

Gets the active\_thread\_count of this ReportingTaskDTO. The number of active threads for the reporting task.

**Returns** The active\_thread\_count of this ReportingTaskDTO.

**Return type** int

#### **annotation\_data**

Gets the annotation\_data of this ReportingTaskDTO. The annotation data for the repoting task. This is how the custom UI relays configuration to the reporting task.

**Returns** The annotation\_data of this ReportingTaskDTO.

**Return type** str

**attribute\_map** = {'active\_thread\_count': 'activeThreadCount', 'annotation\_data': 'ann

**bundle**

Gets the bundle of this ReportingTaskDTO. The details of the artifact that bundled this processor type.

**Returns** The bundle of this ReportingTaskDTO.

**Return type** *BundleDTO*

**comments**

Gets the comments of this ReportingTaskDTO. The comments of the reporting task.

**Returns** The comments of this ReportingTaskDTO.

**Return type** str

**custom\_ui\_url**

Gets the custom\_ui\_url of this ReportingTaskDTO. The URL for the custom configuration UI for the reporting task.

**Returns** The custom\_ui\_url of this ReportingTaskDTO.

**Return type** str

**default\_scheduling\_period**

Gets the default\_scheduling\_period of this ReportingTaskDTO. The default scheduling period for the different scheduling strategies.

**Returns** The default\_scheduling\_period of this ReportingTaskDTO.

**Return type** dict(str, str)

**deprecated**

Gets the deprecated of this ReportingTaskDTO. Whether the reporting task has been deprecated.

**Returns** The deprecated of this ReportingTaskDTO.

**Return type** bool

**descriptors**

Gets the descriptors of this ReportingTaskDTO. The descriptors for the reporting tasks properties.

**Returns** The descriptors of this ReportingTaskDTO.

**Return type** dict(str, *PropertyDescriptorDTO*)

**extension\_missing**

Gets the extension\_missing of this ReportingTaskDTO. Whether the underlying extension is missing.

**Returns** The extension\_missing of this ReportingTaskDTO.

**Return type** bool

**id**

Gets the id of this ReportingTaskDTO. The id of the component.

**Returns** The id of this ReportingTaskDTO.

**Return type** str

**multiple\_versions\_available**

Gets the multiple\_versions\_available of this ReportingTaskDTO. Whether the reporting task has multiple versions available.

**Returns** The multiple\_versions\_available of this ReportingTaskDTO.

**Return type** bool

**name**

Gets the name of this ReportingTaskDTO. The name of the reporting task.

**Returns** The name of this ReportingTaskDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this ReportingTaskDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this ReportingTaskDTO.

**Return type** str

**persists\_state**

Gets the persists\_state of this ReportingTaskDTO. Whether the reporting task persists state.

**Returns** The persists\_state of this ReportingTaskDTO.

**Return type** bool

**position**

Gets the position of this ReportingTaskDTO. The position of this component in the UI if applicable.

**Returns** The position of this ReportingTaskDTO.

**Return type** *PositionDTO*

**properties**

Gets the properties of this ReportingTaskDTO. The properties of the reporting task.

**Returns** The properties of this ReportingTaskDTO.

**Return type** dict(str, str)

**restricted**

Gets the restricted of this ReportingTaskDTO. Whether the reporting task requires elevated privileges.

**Returns** The restricted of this ReportingTaskDTO.

**Return type** bool

**scheduling\_period**

Gets the scheduling\_period of this ReportingTaskDTO. The frequency with which to schedule the reporting task. The format of the value will depend on the value of the schedulingStrategy.

**Returns** The scheduling\_period of this ReportingTaskDTO.

**Return type** str

**scheduling\_strategy**

Gets the scheduling\_strategy of this ReportingTaskDTO. The scheduling strategy that determines how the schedulingPeriod value should be interpreted.

**Returns** The scheduling\_strategy of this ReportingTaskDTO.

**Return type** str

**sensitive\_dynamic\_property\_names**

Gets the sensitive\_dynamic\_property\_names of this ReportingTaskDTO. Set of sensitive dynamic property names

**Returns** The sensitive\_dynamic\_property\_names of this ReportingTaskDTO.

**Return type** list[str]

**state**

Gets the state of this ReportingTaskDTO. The state of the reporting task.

**Returns** The state of this ReportingTaskDTO.

**Return type** str

**supports\_sensitive\_dynamic\_properties**

Gets the supports\_sensitive\_dynamic\_properties of this ReportingTaskDTO. Whether the reporting task supports sensitive dynamic properties.

**Returns** The supports\_sensitive\_dynamic\_properties of this ReportingTaskDTO.

**Return type** bool

**swagger\_types** = {'active\_thread\_count': 'int', 'annotation\_data': 'str', 'bundle':

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ReportingTaskDTO. The fully qualified type of the reporting task.

**Returns** The type of this ReportingTaskDTO.

**Return type** str

**validation\_errors**

Gets the validation\_errors of this ReportingTaskDTO. Gets the validation errors from the reporting task. These validation errors represent the problems with the reporting task that must be resolved before it can be scheduled to run.

**Returns** The validation\_errors of this ReportingTaskDTO.

**Return type** list[str]

**validation\_status**

Gets the validation\_status of this ReportingTaskDTO. Indicates whether the Processor is valid, invalid, or still in the process of validating (i.e., it is unknown whether or not the Processor is valid)

**Returns** The validation\_status of this ReportingTaskDTO.

**Return type** str

**versioned\_component\_id**

Gets the versioned\_component\_id of this ReportingTaskDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this ReportingTaskDTO.

**Return type** str

## nipyapi.nifi.models.reporting\_task\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.reporting_task_entity.ReportingTaskEntity (revision=None,
                                                                    id=None,
                                                                    uri=None,
                                                                    posi-
                                                                    tion=None,
                                                                    permis-
                                                                    sions=None,
                                                                    bul-
                                                                    letins=None,
                                                                    discon-
                                                                    nected_node_acknowledged=None,
                                                                    compo-
                                                                    nent=None,
                                                                    oper-
                                                                    ate_permissions=None,
                                                                    sta-
                                                                    tus=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ReportingTaskEntity - a model defined in Swagger

```
attribute_map = {'bulletins': 'bulletins', 'component': 'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this ReportingTaskEntity. The bulletins for this component.

**Returns** The bulletins of this ReportingTaskEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this ReportingTaskEntity.

**Returns** The component of this ReportingTaskEntity.

**Return type** *ReportingTaskDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this ReportingTaskEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this ReportingTaskEntity.

**Return type** bool

**id**

Gets the id of this ReportingTaskEntity. The id of the component.

**Returns** The id of this ReportingTaskEntity.

**Return type** str

**operate\_permissions**

Gets the operate\_permissions of this ReportingTaskEntity. The permissions for this component operations.

**Returns** The operate\_permissions of this ReportingTaskEntity.

**Return type** *PermissionsDTO*

**permissions**

Gets the permissions of this ReportingTaskEntity. The permissions for this component.

**Returns** The permissions of this ReportingTaskEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this ReportingTaskEntity. The position of this component in the UI if applicable.

**Returns** The position of this ReportingTaskEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this ReportingTaskEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this ReportingTaskEntity.

**Return type** *RevisionDTO*

**status**

Gets the status of this ReportingTaskEntity. The status for this ReportingTask.

**Returns** The status of this ReportingTaskEntity.

**Return type** ReportingTaskStatusDTO

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'ReportingTaskDTO

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this ReportingTaskEntity. The URI for futures requests to the component.

**Returns** The uri of this ReportingTaskEntity.

**Return type** str

## nipyapi.nifi.models.reporting\_task\_types\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.reporting\_task\_types\_entity.**ReportingTaskTypesEntity**(reporting\_task\_t  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ReportingTaskTypesEntity - a model defined in Swagger

```

attribute_map = {'reporting_task_types': 'reportingTaskTypes'}

reporting_task_types
    Gets the reporting_task_types of this ReportingTaskTypesEntity.

    Returns The reporting_task_types of this ReportingTaskTypesEntity.

    Return type list[DocumentedTypeDTO]

swagger_types = {'reporting_task_types': 'list[DocumentedTypeDTO]'}

to_dict()
    Returns the model properties as a dict

to_str()
    Returns the string representation of the model

```

### nipyapi.nifi.models.reporting\_tasks\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.reporting_tasks_entity.ReportingTasksEntity(reporting_tasks=None)
    Bases: object

```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ReportingTasksEntity - a model defined in Swagger

```

attribute_map = {'reporting_tasks': 'reportingTasks'}

reporting_tasks
    Gets the reporting_tasks of this ReportingTasksEntity.

    Returns The reporting_tasks of this ReportingTasksEntity.

    Return type list[ReportingTaskEntity]

swagger_types = {'reporting_tasks': 'list[ReportingTaskEntity]'}

to_dict()
    Returns the model properties as a dict

to_str()
    Returns the string representation of the model

```

### nipyapi.nifi.models.resource\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.resource_dto.ResourceDTO (identifier=None, name=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    ResourceDTO - a model defined in Swagger

    attribute_map = {'identifier': 'identifier', 'name': 'name'}

    identifier
        Gets the identifier of this ResourceDTO. The identifier of the resource.

        Returns The identifier of this ResourceDTO.

        Return type str

    name
        Gets the name of this ResourceDTO. The name of the resource.

        Returns The name of this ResourceDTO.

        Return type str

    swagger_types = {'identifier': 'str', 'name': 'str'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model
```

## nipyapi.nifi.models.resources\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.resources_entity.ResourcesEntity (resources=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    ResourcesEntity - a model defined in Swagger

    attribute_map = {'resources': 'resources'}

    resources
        Gets the resources of this ResourcesEntity.

        Returns The resources of this ResourcesEntity.

        Return type list[ResourceDTO]

    swagger_types = {'resources': 'list[ResourceDTO]'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model
```



## nipyapi.nifi.models.revision\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.revision_dto.RevisionDTO (client_id=None, version=None,  
                                                    last_modifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

RevisionDTO - a model defined in Swagger

```
attribute_map = {'client_id': 'clientId', 'last_modifier': 'lastModifier', 'version'
```

```
client_id
```

Gets the client\_id of this RevisionDTO. A client identifier used to make a request. By including a client identifier, the API can allow multiple requests without needing the current revision. Due to the asynchronous nature of requests/responses this was implemented to allow the client to make numerous requests without having to wait for the previous response to come back

**Returns** The client\_id of this RevisionDTO.

**Return type** str

```
last_modifier
```

Gets the last\_modifier of this RevisionDTO. The user that last modified the flow.

**Returns** The last\_modifier of this RevisionDTO.

**Return type** str

```
swagger_types = {'client_id': 'str', 'last_modifier': 'str', 'version': 'int'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

```
version
```

Gets the version of this RevisionDTO. NiFi employs an optimistic locking strategy where the client must include a revision in their request when performing an update. In a response to a mutable flow request, this field represents the updated base version.

**Returns** The version of this RevisionDTO.

**Return type** int

## nipyapi.nifi.models.schedule\_components\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.schedule_components_entity.ScheduleComponentsEntity (id=None,  
state=None,  
components=None,  
disconnected_node_acked=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ScheduleComponentsEntity - a model defined in Swagger

```
attribute_map = {'components': 'components', 'disconnected_node_acked': 'disconnected_node_acked'}
```

**components**

Gets the components of this ScheduleComponentsEntity. Optional components to schedule. If not specified, all authorized descendant components will be used.

**Returns** The components of this ScheduleComponentsEntity.

**Return type** dict(str, *RevisionDTO*)

**disconnected\_node\_acked**

Gets the disconnected\_node\_acked of this ScheduleComponentsEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acked of this ScheduleComponentsEntity.

**Return type** bool

**id**

Gets the id of this ScheduleComponentsEntity. The id of the ProcessGroup

**Returns** The id of this ScheduleComponentsEntity.

**Return type** str

**state**

Gets the state of this ScheduleComponentsEntity. The desired state of the descendant components

**Returns** The state of this ScheduleComponentsEntity.

**Return type** str

```
swagger_types = {'components': 'dict(str, RevisionDTO)', 'disconnected_node_acked': 'bool'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## **nipyapi.nifi.models.search\_results\_dto module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.search_results_dto.SearchResultsDTO (processor_results=None,
                                                                connec-
                                                                tion_results=None,
                                                                pro-
                                                                cess_group_results=None,
                                                                in-
                                                                put_port_results=None,
                                                                out-
                                                                put_port_results=None,
                                                                re-
                                                                mote_process_group_results=None,
                                                                fun-
                                                                nel_results=None,
                                                                la-
                                                                bel_results=None,
                                                                con-
                                                                troller_service_node_results=None,
                                                                parame-
                                                                ter_context_results=None,
                                                                parame-
                                                                ter_results=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SearchResultsDTO - a model defined in Swagger

```
attribute_map = {'connection_results': 'connectionResults', 'controller_service_node_results': 'controllerServiceNodeResults'}
```

#### **connection\_results**

Gets the connection\_results of this SearchResultsDTO. The connections that matched the search.

**Returns** The connection\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

#### **controller\_service\_node\_results**

Gets the controller\_service\_node\_results of this SearchResultsDTO. The controller service nodes that matched the search

**Returns** The controller\_service\_node\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

#### **funnel\_results**

Gets the funnel\_results of this SearchResultsDTO. The funnels that matched the search.

**Returns** The funnel\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

#### **input\_port\_results**

Gets the input\_port\_results of this SearchResultsDTO. The input ports that matched the search.

**Returns** The input\_port\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**label\_results**

Gets the label\_results of this SearchResultsDTO. The labels that matched the search.

**Returns** The label\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**output\_port\_results**

Gets the output\_port\_results of this SearchResultsDTO. The output ports that matched the search.

**Returns** The output\_port\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**parameter\_context\_results**

Gets the parameter\_context\_results of this SearchResultsDTO. The parameter contexts that matched the search.

**Returns** The parameter\_context\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**parameter\_results**

Gets the parameter\_results of this SearchResultsDTO. The parameters that matched the search.

**Returns** The parameter\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**process\_group\_results**

Gets the process\_group\_results of this SearchResultsDTO. The process groups that matched the search.

**Returns** The process\_group\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**processor\_results**

Gets the processor\_results of this SearchResultsDTO. The processors that matched the search.

**Returns** The processor\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**remote\_process\_group\_results**

Gets the remote\_process\_group\_results of this SearchResultsDTO. The remote process groups that matched the search.

**Returns** The remote\_process\_group\_results of this SearchResultsDTO.

**Return type** list[*ComponentSearchResultDTO*]

**swagger\_types** = {'connection\_results': 'list[ComponentSearchResultDTO]', 'controller\_

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.search\_results\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.search_results_entity.SearchResultsEntity(search_results_dto=None)
    Bases: object
```

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SearchResultsEntity - a model defined in Swagger

```
attribute_map = {'search_results_dto': 'searchResultsDTO'}
```

```
search_results_dto
```

Gets the search\_results\_dto of this SearchResultsEntity.

**Returns** The search\_results\_dto of this SearchResultsEntity.

**Return type** *SearchResultsDTO*

```
swagger_types = {'search_results_dto': 'SearchResultsDTO'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

### nipyapi.nifi.models.snippet\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.snippet_dto.SnippetDTO(id=None, uri=None, parent_group_id=None, process_groups=None, remote_process_groups=None, processors=None, input_ports=None, output_ports=None, connections=None, labels=None, funnels=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SnippetDTO - a model defined in Swagger

```
attribute_map = {'connections': 'connections', 'funnels': 'funnels', 'id': 'id', 'input_ports': 'input_ports', 'output_ports': 'output_ports', 'parent_group_id': 'parent_group_id', 'process_groups': 'process_groups', 'remote_process_groups': 'remote_process_groups', 'processors': 'processors', 'uri': 'uri', 'labels': 'labels', 'funnels': 'funnels'}
```

```
connections
```

Gets the connections of this SnippetDTO. The ids of the connections in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The connections of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **funnels**

Gets the funnels of this SnippetDTO. The ids of the funnels in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The funnels of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **id**

Gets the id of this SnippetDTO. The id of the snippet.

**Returns** The id of this SnippetDTO.

**Return type** str

#### **input\_ports**

Gets the input\_ports of this SnippetDTO. The ids of the input ports in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The input\_ports of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **labels**

Gets the labels of this SnippetDTO. The ids of the labels in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The labels of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **output\_ports**

Gets the output\_ports of this SnippetDTO. The ids of the output ports in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The output\_ports of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **parent\_group\_id**

Gets the parent\_group\_id of this SnippetDTO. The group id for the components in the snippet.

**Returns** The parent\_group\_id of this SnippetDTO.

**Return type** str

#### **process\_groups**

Gets the process\_groups of this SnippetDTO. The ids of the process groups in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The process\_groups of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **processors**

Gets the processors of this SnippetDTO. The ids of the processors in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The processors of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

#### **remote\_process\_groups**

Gets the remote\_process\_groups of this SnippetDTO. The ids of the remote process groups in this snippet. These ids will be populated within each response. They can be specified when creating a snippet. However, once a snippet has been created its contents cannot be modified (these ids are ignored during update requests).

**Returns** The remote\_process\_groups of this SnippetDTO.

**Return type** dict(str, *RevisionDTO*)

**swagger\_types** = {'connections': 'dict(str, RevisionDTO)', 'funnels': 'dict(str, Revi

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this SnippetDTO. The URI of the snippet.

**Returns** The uri of this SnippetDTO.

**Return type** str

### **nipyapi.nifi.models.snippet\_entity module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.snippet\_entity.**SnippetEntity** (snippet=None, disconnected\_node\_acknowledged=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SnippetEntity - a model defined in Swagger

**attribute\_map** = {'disconnected\_node\_acknowledged': 'disconnectedNodeAcknowledged', 's

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this SnippetEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this SnippetEntity.

**Return type** bool

**snippet**

Gets the snippet of this SnippetEntity. The snippet.

**Returns** The snippet of this SnippetEntity.

**Return type** *SnippetDTO*

```
swagger_types = {'disconnected_node_acknowledged': 'bool', 'snippet': 'SnippetDTO'}
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
```

## nipyapi.nifi.models.start\_version\_control\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.start_version_control_request_entity.StartVersionControlRequestEntity
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StartVersionControlRequestEntity - a model defined in Swagger

```
attribute_map = {'disconnected_node_acknowledged': 'disconnectedNodeAcknowledged', 'p
```

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this StartVersionControlRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this StartVersionControlRequestEntity.

**Return type** bool

**process\_group\_revision**

Gets the process\_group\_revision of this StartVersionControlRequestEntity. The Revision of the Process Group under Version Control

**Returns** The process\_group\_revision of this StartVersionControlRequestEntity.

**Return type** *RevisionDTO*

```
swagger_types = {'disconnected_node_acknowledged': 'bool', 'process_group_revision':
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_flow**

Gets the versioned\_flow of this StartVersionControlRequestEntity. The versioned flow

**Returns** The versioned\_flow of this StartVersionControlRequestEntity.

**Return type** *VersionedFlowDTO*



## nipyapi.nifi.models.state\_entry\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.state_entry_dto.StateEntryDTO (key=None,
                                                         value=None,          clus-
                                                         ter_node_id=None, clus-
                                                         ter_node_address=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StateEntryDTO - a model defined in Swagger

```
attribute_map = {'cluster_node_address': 'clusterNodeAddress', 'cluster_node_id': 'c
```

```
cluster_node_address
```

Gets the cluster\_node\_address of this StateEntryDTO. The label for the node where the state originated.

**Returns** The cluster\_node\_address of this StateEntryDTO.

**Return type** str

```
cluster_node_id
```

Gets the cluster\_node\_id of this StateEntryDTO. The identifier for the node where the state originated.

**Returns** The cluster\_node\_id of this StateEntryDTO.

**Return type** str

```
key
```

Gets the key of this StateEntryDTO. The key for this state.

**Returns** The key of this StateEntryDTO.

**Return type** str

```
swagger_types = {'cluster_node_address': 'str', 'cluster_node_id': 'str', 'key': 's
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

```
value
```

Gets the value of this StateEntryDTO. The value for this state.

**Returns** The value of this StateEntryDTO.

**Return type** str

## nipyapi.nifi.models.state\_map\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.state_map_dto.StateMapDTO (scope=None,          to-  
                                                    tal_entry_count=None,  
                                                    state=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StateMapDTO - a model defined in Swagger

```
attribute_map = {'scope':  'scope', 'state':  'state', 'total_entry_count':  'totalEnt
```

**scope**

Gets the scope of this StateMapDTO. The scope of this StateMap.

**Returns** The scope of this StateMapDTO.

**Return type** str

**state**

Gets the state of this StateMapDTO. The state.

**Returns** The state of this StateMapDTO.

**Return type** list[StateEntryDTO]

```
swagger_types = {'scope':  'str', 'state':  'list[StateEntryDTO]', 'total_entry_count'
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**total\_entry\_count**

Gets the total\_entry\_count of this StateMapDTO. The total number of state entries. When the state map is lengthy, only of portion of the entries are returned.

**Returns** The total\_entry\_count of this StateMapDTO.

**Return type** int

## **nipyapi.nifi.models.status\_descriptor\_dto module**

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.status_descriptor_dto.StatusDescriptorDTO (field=None,
                                                                    la-
                                                                    bel=None,
                                                                    descrip-
                                                                    tion=None,
                                                                    format-
                                                                    ter=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StatusDescriptorDTO - a model defined in Swagger

```
attribute_map = {'description': 'description', 'field': 'field', 'formatter': 'form
```

**description**

Gets the description of this StatusDescriptorDTO. The description of the status field.

**Returns** The description of this StatusDescriptorDTO.

**Return type** str

**field**

Gets the field of this StatusDescriptorDTO. The name of the status field.

**Returns** The field of this StatusDescriptorDTO.

**Return type** str

**formatter**

Gets the formatter of this StatusDescriptorDTO. The formatter for the status descriptor.

**Returns** The formatter of this StatusDescriptorDTO.

**Return type** str

**label**

Gets the label of this StatusDescriptorDTO. The label for the status field.

**Returns** The label of this StatusDescriptorDTO.

**Return type** str

```
swagger_types = {'description': 'str', 'field': 'str', 'formatter': 'str', 'label':
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.status\_history\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.status_history_dto.StatusHistoryDTO(generated=None,  
component_details=None,  
field_descriptors=None,  
aggregate_snapshots=None,  
node_snapshots=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StatusHistoryDTO - a model defined in Swagger

#### **aggregate\_snapshots**

Gets the aggregate\_snapshots of this StatusHistoryDTO. A list of StatusSnapshotDTO objects that provide the actual metric values for the component. If the NiFi instance is clustered, this will represent the aggregate status across all nodes. If the NiFi instance is not clustered, this will represent the status of the entire NiFi instance.

**Returns** The aggregate\_snapshots of this StatusHistoryDTO.

**Return type** list[[StatusSnapshotDTO](#)]

```
attribute_map = {'aggregate_snapshots': 'aggregateSnapshots', 'component_details': 'componentDetails'}
```

#### **component\_details**

Gets the component\_details of this StatusHistoryDTO. A Map of key/value pairs that describe the component that the status history belongs to

**Returns** The component\_details of this StatusHistoryDTO.

**Return type** dict(str, str)

#### **field\_descriptors**

Gets the field\_descriptors of this StatusHistoryDTO. The Descriptors that provide information on each of the metrics provided in the status history

**Returns** The field\_descriptors of this StatusHistoryDTO.

**Return type** list[[StatusDescriptorDTO](#)]

#### **generated**

Gets the generated of this StatusHistoryDTO. When the status history was generated.

**Returns** The generated of this StatusHistoryDTO.

**Return type** str

#### **node\_snapshots**

Gets the node\_snapshots of this StatusHistoryDTO. The NodeStatusSnapshotsDTO objects that provide the actual metric values for the component, for each node. If the NiFi instance is not clustered, this value will be null.

**Returns** The node\_snapshots of this StatusHistoryDTO.

**Return type** list[[NodeStatusSnapshotsDTO](#)]

```
swagger_types = {'aggregate_snapshots': 'list[StatusSnapshotDTO]', 'component_details': 'dict[str, str]'
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.status\_history\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.status_history_entity.StatusHistoryEntity(status_history=None,
                                                                    can_read=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StatusHistoryEntity - a model defined in Swagger

```
attribute_map = {'can_read':  'canRead', 'status_history':  'statusHistory'}
```

**can\_read**

Gets the can\_read of this StatusHistoryEntity. Indicates whether the user can read a given resource.

**Returns** The can\_read of this StatusHistoryEntity.

**Return type** bool

**status\_history**

Gets the status\_history of this StatusHistoryEntity.

**Returns** The status\_history of this StatusHistoryEntity.

**Return type** *StatusHistoryDTO*

```
swagger_types = {'can_read':  'bool', 'status_history':  'StatusHistoryDTO'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.status\_snapshot\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.status_snapshot_dto.StatusSnapshotDTO(timestamp=None,
                                                                    sta-
                                                                    tus_metrics=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StatusSnapshotDTO - a model defined in Swagger

```
attribute_map = {'status_metrics':  'statusMetrics', 'timestamp':  'timestamp'}
```

**status\_metrics**

Gets the status\_metrics of this StatusSnapshotDTO. The status metrics.

**Returns** The status\_metrics of this StatusSnapshotDTO.

**Return type** dict(str, int)

**swagger\_types** = {'status\_metrics': 'dict(str, int)', 'timestamp': 'datetime'}

**timestamp**

Gets the timestamp of this StatusSnapshotDTO. The timestamp of the snapshot.

**Returns** The timestamp of this StatusSnapshotDTO.

**Return type** datetime

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.storage\_usage\_dto module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.storage_usage_dto.StorageUsageDTO(identifier=None,  
free_space=None,  
total_space=None,  
used_space=None,  
free_space_bytes=None,  
to-  
tal_space_bytes=None,  
used_space_bytes=None,  
utilization=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StorageUsageDTO - a model defined in Swagger

**attribute\_map** = {'free\_space': 'freeSpace', 'free\_space\_bytes': 'freeSpaceBytes', 'i

**free\_space**

Gets the free\_space of this StorageUsageDTO. Amount of free space.

**Returns** The free\_space of this StorageUsageDTO.

**Return type** str

**free\_space\_bytes**

Gets the free\_space\_bytes of this StorageUsageDTO. The number of bytes of free space.

**Returns** The free\_space\_bytes of this StorageUsageDTO.

**Return type** int

**identifier**

Gets the identifier of this StorageUsageDTO. The identifier of this storage location. The identifier will correspond to the identifier keyed in the storage configuration.

**Returns** The identifier of this StorageUsageDTO.

**Return type** str

```
swagger_types = {'free_space': 'str', 'free_space_bytes': 'int', 'identifier': 'str'
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**total\_space**

Gets the total\_space of this StorageUsageDTO. Amount of total space.

**Returns** The total\_space of this StorageUsageDTO.

**Return type** str

**total\_space\_bytes**

Gets the total\_space\_bytes of this StorageUsageDTO. The number of bytes of total space.

**Returns** The total\_space\_bytes of this StorageUsageDTO.

**Return type** int

**used\_space**

Gets the used\_space of this StorageUsageDTO. Amount of used space.

**Returns** The used\_space of this StorageUsageDTO.

**Return type** str

**used\_space\_bytes**

Gets the used\_space\_bytes of this StorageUsageDTO. The number of bytes of used space.

**Returns** The used\_space\_bytes of this StorageUsageDTO.

**Return type** int

**utilization**

Gets the utilization of this StorageUsageDTO. Utilization of this storage location.

**Returns** The utilization of this StorageUsageDTO.

**Return type** str

## nipyapi.nifi.models.streaming\_output module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.streaming_output.StreamingOutput
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

StreamingOutput - a model defined in Swagger

```
attribute_map = {}
```

```
swagger_types = {}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.submit\_replay\_request\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.submit_replay_request_entity.SubmitReplayRequestEntity(event_id=None, clus-
```

*ter\_node\_id=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SubmitReplayRequestEntity - a model defined in Swagger

```
attribute_map = {'cluster_node_id': 'clusterNodeId', 'event_id': 'eventId'}
```

```
cluster_node_id
```

Gets the cluster\_node\_id of this SubmitReplayRequestEntity. The identifier of the node where to submit the replay request.

**Returns** The cluster\_node\_id of this SubmitReplayRequestEntity.

**Return type** str

```
event_id
```

Gets the event\_id of this SubmitReplayRequestEntity. The event identifier

**Returns** The event\_id of this SubmitReplayRequestEntity.

**Return type** int

```
swagger_types = {'cluster_node_id': 'str', 'event_id': 'int'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model



## nipyapi.nifi.models.system\_diagnostics\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.system\_diagnostics\_dto.**SystemDiagnosticsDTO** (*aggregate\_snapshot=None, node\_snapshots=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SystemDiagnosticsDTO - a model defined in Swagger

#### **aggregate\_snapshot**

Gets the aggregate\_snapshot of this SystemDiagnosticsDTO. A systems diagnostic snapshot that represents the aggregate values of all nodes in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this represents the stats of the single instance.

**Returns** The aggregate\_snapshot of this SystemDiagnosticsDTO.

**Return type** *SystemDiagnosticsSnapshotDTO*

**attribute\_map** = {'aggregate\_snapshot': 'aggregateSnapshot', 'node\_snapshots': 'nodeSnapshots'}

#### **node\_snapshots**

Gets the node\_snapshots of this SystemDiagnosticsDTO. A systems diagnostics snapshot for each node in the cluster. If the NiFi instance is a standalone instance, rather than a cluster, this may be null.

**Returns** The node\_snapshots of this SystemDiagnosticsDTO.

**Return type** list[*NodeSystemDiagnosticsSnapshotDTO*]

**swagger\_types** = {'aggregate\_snapshot': 'SystemDiagnosticsSnapshotDTO', 'node\_snapshots': 'list[SystemDiagnosticsSnapshotDTO]'}  
**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.system\_diagnostics\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.system\_diagnostics\_entity.**SystemDiagnosticsEntity** (*system\_diagnostics=SystemDiagnosticsDTO*)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SystemDiagnosticsEntity - a model defined in Swagger

```
attribute_map = {'system_diagnostics': 'systemDiagnostics'}
swagger_types = {'system_diagnostics': 'SystemDiagnosticsDTO'}
```

**system\_diagnostics**

Gets the system\_diagnostics of this SystemDiagnosticsEntity.

**Returns** The system\_diagnostics of this SystemDiagnosticsEntity.

**Return type** *SystemDiagnosticsDTO*

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO (total_
to-
tal_n
used_
used_
used_
free_
free_
max_
max_
non_
to-
tal_h
to-
tal_h
used_
used_
free_
free_
max_
max_
heap_
avail_
able_
pro-
ces-
sor_l
to-
tal_t
dae-
mon_
up-
time_
flow_
con-
tent_
proce
nanc
garb
stats
ver-
sion_

```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

SystemDiagnosticsSnapshotDTO - a model defined in Swagger

```
attribute_map = {'available_processors': 'availableProcessors', 'content_repository_s
```

**available\_processors**

Gets the available\_processors of this SystemDiagnosticsSnapshotDTO. Number of available processors if supported by the underlying system.

**Returns** The available\_processors of this SystemDiagnosticsSnapshotDTO.

**Return type** int

```
content_repository_storage_usage
```

Gets the `content_repository_storage_usage` of this `SystemDiagnosticsSnapshotDTO`. The content repository storage usage.

**Returns** The `content_repository_storage_usage` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** list[*StorageUsageDTO*]

#### **daemon\_threads**

Gets the `daemon_threads` of this `SystemDiagnosticsSnapshotDTO`. Number of daemon threads.

**Returns** The `daemon_threads` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** int

#### **flow\_file\_repository\_storage\_usage**

Gets the `flow_file_repository_storage_usage` of this `SystemDiagnosticsSnapshotDTO`. The flowfile repository storage usage.

**Returns** The `flow_file_repository_storage_usage` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** *StorageUsageDTO*

#### **free\_heap**

Gets the `free_heap` of this `SystemDiagnosticsSnapshotDTO`. Amount of free heap.

**Returns** The `free_heap` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** str

#### **free\_heap\_bytes**

Gets the `free_heap_bytes` of this `SystemDiagnosticsSnapshotDTO`. The number of bytes that are allocated to the JVM heap but not currently being used

**Returns** The `free_heap_bytes` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** int

#### **free\_non\_heap**

Gets the `free_non_heap` of this `SystemDiagnosticsSnapshotDTO`. Amount of free non heap.

**Returns** The `free_non_heap` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** str

#### **free\_non\_heap\_bytes**

Gets the `free_non_heap_bytes` of this `SystemDiagnosticsSnapshotDTO`. Total number of free non-heap bytes available to the JVM

**Returns** The `free_non_heap_bytes` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** int

#### **garbage\_collection**

Gets the `garbage_collection` of this `SystemDiagnosticsSnapshotDTO`. The garbage collection details.

**Returns** The `garbage_collection` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** list[*GarbageCollectionDTO*]

#### **heap\_utilization**

Gets the `heap_utilization` of this `SystemDiagnosticsSnapshotDTO`. Utilization of heap.

**Returns** The `heap_utilization` of this `SystemDiagnosticsSnapshotDTO`.

**Return type** str

**max\_heap**

Gets the max\_heap of this SystemDiagnosticsSnapshotDTO. Maximum size of heap.

**Returns** The max\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**max\_heap\_bytes**

Gets the max\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. The maximum number of bytes that can be used by the JVM

**Returns** The max\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**max\_non\_heap**

Gets the max\_non\_heap of this SystemDiagnosticsSnapshotDTO. Maximum size of non heap.

**Returns** The max\_non\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**max\_non\_heap\_bytes**

Gets the max\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. The maximum number of bytes that the JVM can use for non-heap purposes

**Returns** The max\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**non\_heap\_utilization**

Gets the non\_heap\_utilization of this SystemDiagnosticsSnapshotDTO. Utilization of non heap.

**Returns** The non\_heap\_utilization of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**processor\_load\_average**

Gets the processor\_load\_average of this SystemDiagnosticsSnapshotDTO. The processor load average if supported by the underlying system.

**Returns** The processor\_load\_average of this SystemDiagnosticsSnapshotDTO.

**Return type** float

**provenance\_repository\_storage\_usage**

Gets the provenance\_repository\_storage\_usage of this SystemDiagnosticsSnapshotDTO. The provenance repository storage usage.

**Returns** The provenance\_repository\_storage\_usage of this SystemDiagnosticsSnapshotDTO.

**Return type** list[[StorageUsageDTO](#)]

**stats\_last\_refreshed**

Gets the stats\_last\_refreshed of this SystemDiagnosticsSnapshotDTO. When the diagnostics were generated.

**Returns** The stats\_last\_refreshed of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**swagger\_types** = {'available\_processors': 'int', 'content\_repository\_storage\_usage':

**to\_dict** ()

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**total\_heap**

Gets the total\_heap of this SystemDiagnosticsSnapshotDTO. Total size of heap.

**Returns** The total\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**total\_heap\_bytes**

Gets the total\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. The total number of bytes that are available for the JVM heap to use

**Returns** The total\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**total\_non\_heap**

Gets the total\_non\_heap of this SystemDiagnosticsSnapshotDTO. Total size of non heap.

**Returns** The total\_non\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**total\_non\_heap\_bytes**

Gets the total\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. Total number of bytes allocated to the JVM not used for heap

**Returns** The total\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**total\_threads**

Gets the total\_threads of this SystemDiagnosticsSnapshotDTO. Total number of threads.

**Returns** The total\_threads of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**uptime**

Gets the uptime of this SystemDiagnosticsSnapshotDTO. The uptime of the Java virtual machine

**Returns** The uptime of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**used\_heap**

Gets the used\_heap of this SystemDiagnosticsSnapshotDTO. Amount of used heap.

**Returns** The used\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

**used\_heap\_bytes**

Gets the used\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. The number of bytes of JVM heap that are currently being used

**Returns** The used\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

**used\_non\_heap**

Gets the used\_non\_heap of this SystemDiagnosticsSnapshotDTO. Amount of use non heap.

**Returns** The used\_non\_heap of this SystemDiagnosticsSnapshotDTO.

**Return type** str

#### **used\_non\_heap\_bytes**

Gets the used\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO. Total number of bytes used by the JVM not in the heap space

**Returns** The used\_non\_heap\_bytes of this SystemDiagnosticsSnapshotDTO.

**Return type** int

#### **version\_info**

Gets the version\_info of this SystemDiagnosticsSnapshotDTO. The nifi, os, java, and build version information

**Returns** The version\_info of this SystemDiagnosticsSnapshotDTO.

**Return type** *VersionInfoDTO*

### **nipyapi.nifi.models.template\_dto module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.template_dto.TemplateDTO (uri=None,          id=None,
                                                    group_id=None,       name=None,
                                                    description=None,      times-
                                                    tamp=None,             encod-
                                                    ing_version=None,      snip-
                                                    pet=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

TemplateDTO - a model defined in Swagger

```
attribute_map = {'description':  'description', 'encoding_version':  'encodingVersion'}
```

#### **description**

Gets the description of this TemplateDTO. The description of the template.

**Returns** The description of this TemplateDTO.

**Return type** str

#### **encoding\_version**

Gets the encoding\_version of this TemplateDTO. The encoding version of this template.

**Returns** The encoding\_version of this TemplateDTO.

**Return type** str

#### **group\_id**

Gets the group\_id of this TemplateDTO. The id of the Process Group that the template belongs to.

**Returns** The group\_id of this TemplateDTO.

**Return type** str

**id**

Gets the id of this TemplateDTO. The id of the template.

**Returns** The id of this TemplateDTO.**Return type** str**name**

Gets the name of this TemplateDTO. The name of the template.

**Returns** The name of this TemplateDTO.**Return type** str**snippet**

Gets the snippet of this TemplateDTO. The contents of the template.

**Returns** The snippet of this TemplateDTO.**Return type** *FlowSnippetDTO***swagger\_types** = {'description': 'str', 'encoding\_version': 'str', 'group\_id': 'str'}**timestamp**

Gets the timestamp of this TemplateDTO. The timestamp when this template was created.

**Returns** The timestamp of this TemplateDTO.**Return type** str**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this TemplateDTO. The URI for the template.

**Returns** The uri of this TemplateDTO.**Return type** str

## nipyapi.nifi.models.template\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.template_entity.TemplateEntity(revision=None, id=None,
                                                         uri=None,           posi-
                                                         tion=None,       permis-
                                                         sions=None,      bul-
                                                         letins=None,     discon-
                                                         nected_node_acknowledged=None,
                                                         template=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.



TemplateEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'disconnected\_node\_acknowledged': 'discon

**bulletins**

Gets the bulletins of this TemplateEntity. The bulletins for this component.

**Returns** The bulletins of this TemplateEntity.

**Return type** list[*BulletinEntity*]

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this TemplateEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this TemplateEntity.

**Return type** bool

**id**

Gets the id of this TemplateEntity. The id of the component.

**Returns** The id of this TemplateEntity.

**Return type** str

**permissions**

Gets the permissions of this TemplateEntity. The permissions for this component.

**Returns** The permissions of this TemplateEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this TemplateEntity. The position of this component in the UI if applicable.

**Returns** The position of this TemplateEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this TemplateEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this TemplateEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'disconnected\_node\_acknowledged

**template**

Gets the template of this TemplateEntity.

**Returns** The template of this TemplateEntity.

**Return type** *TemplateDTO*

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this TemplateEntity. The URI for futures requests to the component.

**Returns** The uri of this TemplateEntity.

**Return type** str

### nipyapi.nifi.models.templates\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.templates_entity.TemplatesEntity (templates=None, generated=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

TemplatesEntity - a model defined in Swagger

```
attribute_map = {'generated': 'generated', 'templates': 'templates'}
```

#### **generated**

Gets the generated of this TemplatesEntity. When this content was generated.

**Returns** The generated of this TemplatesEntity.

**Return type** str

```
swagger_types = {'generated': 'str', 'templates': 'list[TemplateEntity]'}
```

#### **templates**

Gets the templates of this TemplatesEntity.

**Returns** The templates of this TemplatesEntity.

**Return type** list[*TemplateEntity*]

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

### nipyapi.nifi.models.tenant\_dto module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.tenant_dto.TenantDTO (id=None, versioned_component_id=None, parent_group_id=None, position=None, identity=None, configurable=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

TenantDTO - a model defined in Swagger

```
attribute_map = {'configurable': 'configurable', 'id': 'id', 'identity': 'identity'}
```

**configurable**

Gets the configurable of this TenantDTO. Whether this tenant is configurable.

**Returns** The configurable of this TenantDTO.

**Return type** bool

**id**

Gets the id of this TenantDTO. The id of the component.

**Returns** The id of this TenantDTO.

**Return type** str

**identity**

Gets the identity of this TenantDTO. The identity of the tenant.

**Returns** The identity of this TenantDTO.

**Return type** str

**parent\_group\_id**

Gets the parent\_group\_id of this TenantDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this TenantDTO.

**Return type** str

**position**

Gets the position of this TenantDTO. The position of this component in the UI if applicable.

**Returns** The position of this TenantDTO.

**Return type** *PositionDTO*

```
swagger_types = {'configurable': 'bool', 'id': 'str', 'identity': 'str', 'parent_gr
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_component\_id**

Gets the versioned\_component\_id of this TenantDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this TenantDTO.

**Return type** str

### nipyapi.nifi.models.tenant\_entity module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.tenant_entity.TenantEntity (revision=None,      id=None,
                                                    uri=None,      position=None,
                                                    permissions=None,      bul-
                                                    letins=None,      discon-
                                                    nected_node_acknowledged=None,
                                                    component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

TenantEntity - a model defined in Swagger

```
attribute_map = {'bulletins':  'bulletins', 'component':  'component', 'disconnected_n
```

**bulletins**

Gets the bulletins of this TenantEntity. The bulletins for this component.

**Returns** The bulletins of this TenantEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this TenantEntity.

**Returns** The component of this TenantEntity.

**Return type** *TenantDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this TenantEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this TenantEntity.

**Return type** bool

**id**

Gets the id of this TenantEntity. The id of the component.

**Returns** The id of this TenantEntity.

**Return type** str

**permissions**

Gets the permissions of this TenantEntity. The permissions for this component.

**Returns** The permissions of this TenantEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this TenantEntity. The position of this component in the UI if applicable.

**Returns** The position of this TenantEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this TenantEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this TenantEntity.

**Return type** *RevisionDTO*

```

swagger_types = {'bulletins': 'list[BulletinEntity]', 'component': 'TenantDTO', 'dis
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
uri
    Gets the uri of this TenantEntity. The URI for futures requests to the component.

    Returns The uri of this TenantEntity.
    Return type str

```

### nipyapi.nifi.models.tenants\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.nifi.models.tenants_entity.TenantsEntity(users=None,
                                                    user_groups=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    TenantsEntity - a model defined in Swagger

    attribute_map = {'user_groups': 'userGroups', 'users': 'users'}
    swagger_types = {'user_groups': 'list[TenantEntity]', 'users': 'list[TenantEntity]'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model

    user_groups
        Gets the user_groups of this TenantsEntity.

        Returns The user_groups of this TenantsEntity.
        Return type list[TenantEntity]

    users
        Gets the users of this TenantsEntity.

        Returns The users of this TenantsEntity.
        Return type list[TenantEntity]

```

### nipyapi.nifi.models.transaction\_result\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.transaction_result_entity.TransactionResultEntity (flow_file_sent=None,
                                                                              re-
                                                                              sponse_code=None,
                                                                              mes-
                                                                              sage=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

TransactionResultEntity - a model defined in Swagger

```
attribute_map = {'flow_file_sent':  'flowFileSent', 'message':  'message', 'response_code':  'responseCode'}
```

```
flow_file_sent
```

Gets the flow\_file\_sent of this TransactionResultEntity.

**Returns** The flow\_file\_sent of this TransactionResultEntity.

**Return type** int

```
message
```

Gets the message of this TransactionResultEntity.

**Returns** The message of this TransactionResultEntity.

**Return type** str

```
response_code
```

Gets the response\_code of this TransactionResultEntity.

**Returns** The response\_code of this TransactionResultEntity.

**Return type** int

```
swagger_types = {'flow_file_sent':  'int', 'message':  'str', 'response_code':  'int'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

## nipyapi.nifi.models.update\_controller\_service\_reference\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.update_controller_service_reference_request_entity.UpdateControl
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UpdateControllerServiceReferenceRequestEntity - a model defined in Swagger

```
attribute_map = {'disconnected_node_acknowledged': 'disconnectedNodeAcknowledged', 'id': 'id', 'referencing_component_revisions': 'referencingComponentRevisions', 'state': 'state', 'ui_only': 'uiOnly'}
```

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this UpdateControllerServiceReferenceRequestEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this UpdateControllerServiceReferenceRequestEntity.

**Return type** bool

**id**

Gets the id of this UpdateControllerServiceReferenceRequestEntity. The identifier of the Controller Service.

**Returns** The id of this UpdateControllerServiceReferenceRequestEntity.

**Return type** str

**referencing\_component\_revisions**

Gets the referencing\_component\_revisions of this UpdateControllerServiceReferenceRequestEntity. The revisions for all referencing components.

**Returns** The referencing\_component\_revisions of this UpdateControllerServiceReferenceRequestEntity.

**Return type** dict(str, *RevisionDTO*)

**state**

Gets the state of this UpdateControllerServiceReferenceRequestEntity. The new state of the references for the controller service.

**Returns** The state of this UpdateControllerServiceReferenceRequestEntity.

**Return type** str

```
swagger_types = {'disconnected_node_acknowledged': 'bool', 'id': 'str', 'referencing_component_revisions': 'dict(str, RevisionDTO)', 'state': 'str', 'ui_only': 'bool'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**ui\_only**

Gets the ui\_only of this UpdateControllerServiceReferenceRequestEntity. Indicates whether or not the response should only include fields necessary for rendering the NiFi User Interface. As such, when this

value is set to true, some fields may be returned as null values, and the selected fields may change at any time without notice. As a result, this value should not be set to true by any client other than the UI.

**Returns** The ui\_only of this UpdateControllerServiceReferenceRequestEntity.

**Return type** bool

## nipyapi.nifi.models.user\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.user_dto.UserDTO (id=None, versioned_component_id=None,  
                                             parent_group_id=None, position=None,  
                                             identity=None, configurable=None,  
                                             user_groups=None, access_policies=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UserDTO - a model defined in Swagger

#### access\_policies

Gets the access\_policies of this UserDTO. The access policies this user belongs to.

**Returns** The access\_policies of this UserDTO.

**Return type** list[*AccessPolicySummaryEntity*]

```
attribute_map = {'access_policies': 'accessPolicies', 'configurable': 'configurable'}
```

#### configurable

Gets the configurable of this UserDTO. Whether this tenant is configurable.

**Returns** The configurable of this UserDTO.

**Return type** bool

#### id

Gets the id of this UserDTO. The id of the component.

**Returns** The id of this UserDTO.

**Return type** str

#### identity

Gets the identity of this UserDTO. The identity of the tenant.

**Returns** The identity of this UserDTO.

**Return type** str

#### parent\_group\_id

Gets the parent\_group\_id of this UserDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this UserDTO.

**Return type** str



**position**

Gets the position of this UserDTO. The position of this component in the UI if applicable.

**Returns** The position of this UserDTO.

**Return type** *PositionDTO*

**swagger\_types** = {'access\_policies': 'list[AccessPolicySummaryEntity]', 'configurable'

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**user\_groups**

Gets the user\_groups of this UserDTO. The groups to which the user belongs. This field is read only and it provided for convenience.

**Returns** The user\_groups of this UserDTO.

**Return type** list[*TenantEntity*]

**versioned\_component\_id**

Gets the versioned\_component\_id of this UserDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this UserDTO.

**Return type** str

**nipyapi.nifi.models.user\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.user_entity.UserEntity (revision=None, id=None,
                                                    uri=None, position=None, permis-
                                                    sions=None, bulletins=None, discon-
                                                    nected_node_acknowledged=None,
                                                    component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UserEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'component': 'component', 'disconnected\_n

**bulletins**

Gets the bulletins of this UserEntity. The bulletins for this component.

**Returns** The bulletins of this UserEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this UserEntity.

**Returns** The component of this UserEntity.

**Return type** *UserDTO*

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this UserEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this UserEntity.

**Return type** bool

**id**

Gets the id of this UserEntity. The id of the component.

**Returns** The id of this UserEntity.

**Return type** str

**permissions**

Gets the permissions of this UserEntity. The permissions for this component.

**Returns** The permissions of this UserEntity.

**Return type** *PermissionsDTO*

**position**

Gets the position of this UserEntity. The position of this component in the UI if applicable.

**Returns** The position of this UserEntity.

**Return type** *PositionDTO*

**revision**

Gets the revision of this UserEntity. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this UserEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'UserDTO', 'disconnected\_node\_acknowledged': 'bool', 'id': 'str', 'permissions': 'PermissionsDTO', 'position': 'PositionDTO', 'revision': 'RevisionDTO', 'uri': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this UserEntity. The URI for futures requests to the component.

**Returns** The uri of this UserEntity.

**Return type** str

## nipyapi.nifi.models.user\_group\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.user_group_dto.UserGroupDTO (id=None, versioned_component_id=None, parent_group_id=None, position=None, identity=None, configurable=None, users=None, access_policies=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UserGroupDTO - a model defined in Swagger

#### **access\_policies**

Gets the access\_policies of this UserGroupDTO. The access policies this user group belongs to. This field was incorrectly defined as an AccessPolicyEntity. For compatibility reasons the field will remain of this type, however only the fields that are present in the AccessPolicySummaryEntity will be populated here.

**Returns** The access\_policies of this UserGroupDTO.

**Return type** list[*AccessPolicyEntity*]

```
attribute_map = {'access_policies': 'accessPolicies', 'configurable': 'configurable'}
```

#### **configurable**

Gets the configurable of this UserGroupDTO. Whether this tenant is configurable.

**Returns** The configurable of this UserGroupDTO.

**Return type** bool

#### **id**

Gets the id of this UserGroupDTO. The id of the component.

**Returns** The id of this UserGroupDTO.

**Return type** str

#### **identity**

Gets the identity of this UserGroupDTO. The identity of the tenant.

**Returns** The identity of this UserGroupDTO.

**Return type** str

#### **parent\_group\_id**

Gets the parent\_group\_id of this UserGroupDTO. The id of parent process group of this component if applicable.

**Returns** The parent\_group\_id of this UserGroupDTO.

**Return type** str

#### **position**

Gets the position of this UserGroupDTO. The position of this component in the UI if applicable.

**Returns** The position of this UserGroupDTO.

**Return type** *PositionDTO*

```
swagger_types = {'access_policies': 'list[AccessPolicyEntity]', 'configurable': 'bool'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**users**

Gets the users of this UserGroupDTO. The users that belong to the user group.

**Returns** The users of this UserGroupDTO.

**Return type** list[*TenantEntity*]

**versioned\_component\_id**

Gets the versioned\_component\_id of this UserGroupDTO. The ID of the corresponding component that is under version control

**Returns** The versioned\_component\_id of this UserGroupDTO.

**Return type** str

## nipyapi.nifi.models.user\_group\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.user_group_entity.UserGroupEntity(revision=None,
                                                             id=None, uri=None,
                                                             position=None, per-
                                                             missions=None, bul-
                                                             letins=None, discon-
                                                             nected_node_acknowledged=None,
                                                             component=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UserGroupEntity - a model defined in Swagger

**attribute\_map** = {'bulletins': 'bulletins', 'component': 'component', 'disconnected\_n

**bulletins**

Gets the bulletins of this UserGroupEntity. The bulletins for this component.

**Returns** The bulletins of this UserGroupEntity.

**Return type** list[*BulletinEntity*]

**component**

Gets the component of this UserGroupEntity.

**Returns** The component of this UserGroupEntity.

**Return type** *UserGroupDTO*

**disconnected\_node\_acknowledged**

Gets the `disconnected_node_acknowledged` of this `UserGroupEntity`. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The `disconnected_node_acknowledged` of this `UserGroupEntity`.

**Return type** `bool`

**id**

Gets the id of this `UserGroupEntity`. The id of the component.

**Returns** The id of this `UserGroupEntity`.

**Return type** `str`

**permissions**

Gets the permissions of this `UserGroupEntity`. The permissions for this component.

**Returns** The permissions of this `UserGroupEntity`.

**Return type** *PermissionsDTO*

**position**

Gets the position of this `UserGroupEntity`. The position of this component in the UI if applicable.

**Returns** The position of this `UserGroupEntity`.

**Return type** *PositionDTO*

**revision**

Gets the revision of this `UserGroupEntity`. The revision for this request/response. The revision is required for any mutable flow requests and is included in all responses.

**Returns** The revision of this `UserGroupEntity`.

**Return type** *RevisionDTO*

**swagger\_types** = {'bulletins': 'list[BulletinEntity]', 'component': 'UserGroupDTO', 'uri': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this `UserGroupEntity`. The URI for futures requests to the component.

**Returns** The uri of this `UserGroupEntity`.

**Return type** `str`

**nipyapi.nifi.models.user\_groups\_entity module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.user_groups_entity.UserGroupsEntity (user_groups=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    UserGroupsEntity - a model defined in Swagger

    attribute_map = {'user_groups': 'userGroups'}

    swagger_types = {'user_groups': 'list[UserGroupEntity]'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model

    user_groups
        Gets the user_groups of this UserGroupsEntity.

        Returns The user_groups of this UserGroupsEntity.

        Return type list[UserGroupEntity]
```

## nipyapi.nifi.models.users\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.users_entity.UsersEntity (generated=None, users=None)
    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    UsersEntity - a model defined in Swagger

    attribute_map = {'generated': 'generated', 'users': 'users'}

    generated
        Gets the generated of this UsersEntity. When this content was generated.

        Returns The generated of this UsersEntity.

        Return type str

    swagger_types = {'generated': 'str', 'users': 'list[UserEntity]'}

    to_dict()
        Returns the model properties as a dict

    to_str()
        Returns the string representation of the model

    users
        Gets the users of this UsersEntity.

        Returns The users of this UsersEntity.

        Return type list[UserEntity]
```

## nipyapi.nifi.models.variable\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.variable_dto.VariableDTO (name=None, value=None,
                                                    process_group_id=None, affected_components=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableDTO - a model defined in Swagger

#### **affected\_components**

Gets the affected\_components of this VariableDTO. A set of all components that will be affected if the value of this variable is changed

**Returns** The affected\_components of this VariableDTO.

**Return type** list[*AffectedComponentEntity*]

```
attribute_map = {'affected_components': 'affectedComponents', 'name': 'name', 'process_group_id': 'processGroupId'}
```

#### **name**

Gets the name of this VariableDTO. The name of the variable

**Returns** The name of this VariableDTO.

**Return type** str

#### **process\_group\_id**

Gets the process\_group\_id of this VariableDTO. The ID of the Process Group where this Variable is defined

**Returns** The process\_group\_id of this VariableDTO.

**Return type** str

```
swagger_types = {'affected_components': 'list[AffectedComponentEntity]', 'name': 'string', 'process_group_id': 'string'}
```

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **value**

Gets the value of this VariableDTO. The value of the variable

**Returns** The value of this VariableDTO.

**Return type** str

## nipyapi.nifi.models.variable\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.variable_entity.VariableEntity (variable=None,  
                                                         can_write=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableEntity - a model defined in Swagger

```
attribute_map = {'can_write': 'canWrite', 'variable': 'variable'}
```

```
can_write
```

Gets the can\_write of this VariableEntity. Indicates whether the user can write a given resource.

**Returns** The can\_write of this VariableEntity.

**Return type** bool

```
swagger_types = {'can_write': 'bool', 'variable': 'VariableDTO'}
```

```
to_dict ()
```

Returns the model properties as a dict

```
to_str ()
```

Returns the string representation of the model

```
variable
```

Gets the variable of this VariableEntity. The variable information

**Returns** The variable of this VariableEntity.

**Return type** *VariableDTO*

## nipyapi.nifi.models.variable\_registry\_dto module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.variable_registry_dto.VariableRegistryDTO (variables=None,  
                                                                    pro-  
                                                                    cess_group_id=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableRegistryDTO - a model defined in Swagger

```
attribute_map = {'process_group_id': 'processGroupId', 'variables': 'variables'}
```

```
process_group_id
```

Gets the process\_group\_id of this VariableRegistryDTO. The UUID of the Process Group that this Variable Registry belongs to



**Returns** The process\_group\_id of this VariableRegistryDTO.

**Return type** str

**swagger\_types** = {'process\_group\_id': 'str', 'variables': 'list[VariableEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**variables**

Gets the variables of this VariableRegistryDTO. The variables that are available in this Variable Registry

**Returns** The variables of this VariableRegistryDTO.

**Return type** list[*VariableEntity*]

### nipyapi.nifi.models.variable\_registry\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.variable\_registry\_entity.**VariableRegistryEntity** (*process\_group\_revision=*  
*variable\_registry=None,*  
*dis-*  
*con-*  
*nected\_node\_acknowledged*

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableRegistryEntity - a model defined in Swagger

**attribute\_map** = {'disconnected\_node\_acknowledged': 'disconnectedNodeAcknowledged', 'p

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this VariableRegistryEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this VariableRegistryEntity.

**Return type** bool

**process\_group\_revision**

Gets the process\_group\_revision of this VariableRegistryEntity. The revision of the Process Group that the Variable Registry belongs to

**Returns** The process\_group\_revision of this VariableRegistryEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'disconnected\_node\_acknowledged': 'bool', 'process\_group\_revision':

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**variable\_registry**

Gets the variable\_registry of this VariableRegistryEntity. The Variable Registry.

**Returns** The variable\_registry of this VariableRegistryEntity.

**Return type** *VariableRegistryDTO*

## nipyapi.nifi.models.variable\_registry\_update\_request\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.variable\_registry\_update\_request\_dto.**VariableRegistryUpdateRequest**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableRegistryUpdateRequestDTO - a model defined in Swagger

**affected\_components**

Gets the affected\_components of this VariableRegistryUpdateRequestDTO. A set of all components that will be affected if the value of this variable is changed

**Returns** The affected\_components of this VariableRegistryUpdateRequestDTO.

**Return type** list[*AffectedComponentEntity*]

**attribute\_map** = {'affected\_components': 'affectedComponents', 'complete': 'complete'}

**complete**

Gets the complete of this VariableRegistryUpdateRequestDTO. Whether or not the request is completed

**Returns** The complete of this VariableRegistryUpdateRequestDTO.

**Return type** bool

**failure\_reason**

Gets the failure\_reason of this VariableRegistryUpdateRequestDTO. The reason for the request failing, or null if the request has not failed

**Returns** The failure\_reason of this VariableRegistryUpdateRequestDTO.

**Return type** str

**last\_updated**

Gets the last\_updated of this VariableRegistryUpdateRequestDTO. The timestamp of when the request was last updated

**Returns** The last\_updated of this VariableRegistryUpdateRequestDTO.

**Return type** datetime

**percent\_completed**

Gets the percent\_completed of this VariableRegistryUpdateRequestDTO. A value between 0 and 100 (inclusive) indicating how close the request is to completion

**Returns** The percent\_completed of this VariableRegistryUpdateRequestDTO.

**Return type** int

**process\_group\_id**

Gets the process\_group\_id of this VariableRegistryUpdateRequestDTO. The unique ID of the Process Group that the variable registry belongs to

**Returns** The process\_group\_id of this VariableRegistryUpdateRequestDTO.

**Return type** str

**request\_id**

Gets the request\_id of this VariableRegistryUpdateRequestDTO. The ID of the request

**Returns** The request\_id of this VariableRegistryUpdateRequestDTO.

**Return type** str

**state**

Gets the state of this VariableRegistryUpdateRequestDTO. A description of the current state of the request

**Returns** The state of this VariableRegistryUpdateRequestDTO.

**Return type** str

**submission\_time**

Gets the submission\_time of this VariableRegistryUpdateRequestDTO. The timestamp of when the request was submitted

**Returns** The submission\_time of this VariableRegistryUpdateRequestDTO.

**Return type** datetime

**swagger\_types** = {'affected\_components': 'list[AffectedComponentEntity]', 'complete':

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**update\_steps**

Gets the update\_steps of this VariableRegistryUpdateRequestDTO. The steps that are required in order to complete the request, along with the status of each

**Returns** The update\_steps of this VariableRegistryUpdateRequestDTO.

**Return type** list[*VariableRegistryUpdateStepDTO*]

**uri**

Gets the uri of this VariableRegistryUpdateRequestDTO. The URI for the request

**Returns** The uri of this VariableRegistryUpdateRequestDTO.

**Return type** str

**nipyapi.nifi.models.variable\_registry\_update\_request\_entity module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.variable\_registry\_update\_request\_entity.**VariableRegistryUpdateRequestEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableRegistryUpdateRequestEntity - a model defined in Swagger

**attribute\_map** = {'process\_group\_revision': 'processGroupRevision', 'request': 'request'}

**process\_group\_revision**

Gets the process\_group\_revision of this VariableRegistryUpdateRequestEntity. The revision for the Process Group that owns this variable registry.

**Returns** The process\_group\_revision of this VariableRegistryUpdateRequestEntity.

**Return type** *RevisionDTO*

**request**

Gets the request of this VariableRegistryUpdateRequestEntity. The Variable Registry Update Request

**Returns** The request of this VariableRegistryUpdateRequestEntity.

**Return type** *VariableRegistryUpdateRequestDTO*

**swagger\_types** = {'process\_group\_revision': 'RevisionDTO', 'request': 'VariableRegistryUpdateRequestDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.variable\_registry\_update\_step\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.variable\_registry\_update\_step\_dto.VariableRegistryUpdateStepDTO (

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VariableRegistryUpdateStepDTO - a model defined in Swagger

**attribute\_map** = {'complete': 'complete', 'description': 'description', 'failure\_reas

**complete**

Gets the complete of this VariableRegistryUpdateStepDTO. Whether or not this step has completed

**Returns** The complete of this VariableRegistryUpdateStepDTO.

**Return type** bool

**description**

Gets the description of this VariableRegistryUpdateStepDTO. Explanation of what happens in this step

**Returns** The description of this VariableRegistryUpdateStepDTO.

**Return type** str

**failure\_reason**

Gets the failure\_reason of this VariableRegistryUpdateStepDTO. An explanation of why this step failed, or null if this step did not fail

**Returns** The failure\_reason of this VariableRegistryUpdateStepDTO.

**Return type** str

**swagger\_types** = {'complete': 'bool', 'description': 'str', 'failure\_reason': 'str'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.version\_control\_component\_mapping\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.version\_control\_component\_mapping\_entity.**VersionControlComponentMappingEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionControlComponentMappingEntity - a model defined in Swagger

**attribute\_map** = {'disconnected\_node\_acknowledged': 'disconnectedNodeAcknowledged', 'process\_group\_revision': 'processGroupRevision'}

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this VersionControlComponentMappingEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this VersionControlComponentMappingEntity.

**Return type** bool

**process\_group\_revision**

Gets the process\_group\_revision of this VersionControlComponentMappingEntity. The revision of the Process Group

**Returns** The process\_group\_revision of this VersionControlComponentMappingEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'disconnected\_node\_acknowledged': 'bool', 'process\_group\_revision': 'RevisionDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**version\_control\_component\_mapping**

Gets the version\_control\_component\_mapping of this VersionControlComponentMappingEntity. The mapping of Versioned Component Identifiers to instance ID's

**Returns** The version\_control\_component\_mapping of this VersionControlComponentMappingEntity.

**Return type** dict(str, str)

**version\_control\_information**

Gets the version\_control\_information of this VersionControlComponentMappingEntity. The Version Control information

**Returns** The version\_control\_information of this VersionControlComponentMappingEntity.

**Return type** *VersionControlInformationDTO*

## nipyapi.nifi.models.version\_control\_information\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO (group

reg-  
istry\_  
reg-  
istry\_  
bucke  
bucke  
flow\_  
flow\_  
flow\_  
ver-  
sion=  
state  
state

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionControlInformationDTO - a model defined in Swagger

**attribute\_map** = {'bucket\_id': 'bucketId', 'bucket\_name': 'bucketName', 'flow\_descrip

**bucket\_id**

Gets the bucket\_id of this VersionControlInformationDTO. The ID of the bucket that the flow is stored in

**Returns** The bucket\_id of this VersionControlInformationDTO.

**Return type** str

**bucket\_name**

Gets the bucket\_name of this VersionControlInformationDTO. The name of the bucket that the flow is stored in

**Returns** The bucket\_name of this VersionControlInformationDTO.

**Return type** str

**flow\_description**

Gets the flow\_description of this VersionControlInformationDTO. The description of the flow

**Returns** The flow\_description of this VersionControlInformationDTO.

**Return type** str

**flow\_id**

Gets the flow\_id of this VersionControlInformationDTO. The ID of the flow

**Returns** The flow\_id of this VersionControlInformationDTO.

**Return type** str

**flow\_name**

Gets the flow\_name of this VersionControlInformationDTO. The name of the flow

**Returns** The flow\_name of this VersionControlInformationDTO.

**Return type** str

**group\_id**

Gets the group\_id of this VersionControlInformationDTO. The ID of the Process Group that is under version control

**Returns** The group\_id of this VersionControlInformationDTO.

**Return type** str

**registry\_id**

Gets the registry\_id of this VersionControlInformationDTO. The ID of the registry that the flow is stored in

**Returns** The registry\_id of this VersionControlInformationDTO.

**Return type** str

**registry\_name**

Gets the registry\_name of this VersionControlInformationDTO. The name of the registry that the flow is stored in

**Returns** The registry\_name of this VersionControlInformationDTO.

**Return type** str

**state**

Gets the state of this VersionControlInformationDTO. The current state of the Process Group, as it relates to the Versioned Flow

**Returns** The state of this VersionControlInformationDTO.

**Return type** str

**state\_explanation**

Gets the state\_explanation of this VersionControlInformationDTO. Explanation of why the group is in the specified state

**Returns** The state\_explanation of this VersionControlInformationDTO.

**Return type** str

**swagger\_types** = {'bucket\_id': 'str', 'bucket\_name': 'str', 'flow\_description': 'str'

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**version**

Gets the version of this VersionControlInformationDTO. The version of the flow

**Returns** The version of this VersionControlInformationDTO.

**Return type** int



## nipyapi.nifi.models.version\_control\_information\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.version\_control\_information\_entity.VersionControlInformationEntity

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionControlInformationEntity - a model defined in Swagger

**attribute\_map** = {'disconnected\_node\_acknowledged': 'disconnectedNodeAcknowledged', 'p

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this VersionControlInformationEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this VersionControlInformationEntity.

**Return type** bool

**process\_group\_revision**

Gets the process\_group\_revision of this VersionControlInformationEntity. The Revision for the Process Group

**Returns** The process\_group\_revision of this VersionControlInformationEntity.

**Return type** *RevisionDTO*

**swagger\_types** = {'disconnected\_node\_acknowledged': 'bool', 'process\_group\_revision':

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**version\_control\_information**

Gets the version\_control\_information of this VersionControlInformationEntity. The Version Control information

**Returns** The version\_control\_information of this VersionControlInformationEntity.

**Return type** *VersionControlInformationDTO*

## nipyapi.nifi.models.version\_info\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.version_info_dto.VersionInfoDTO (ni-fi_version=None,  
                                                         java_vendor=None,  
                                                         java_version=None,  
                                                         os_name=None,  
                                                         os_version=None,  
                                                         os_architecture=None,  
                                                         build_tag=None,  
                                                         build_revision=None,  
                                                         build_branch=None,  
                                                         build_timestamp=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionInfoDTO - a model defined in Swagger

```
attribute_map = {'build_branch':  'buildBranch', 'build_revision':  'buildRevision', 'build_timestamp':  'buildTimestamp', 'java_vendor':  'javaVendor', 'java_version':  'javaVersion', 'os_architecture':  'osArchitecture', 'os_name':  'osName', 'os_version':  'osVersion', 'build_tag':  'buildTag'}
```

**build\_branch**

Gets the build\_branch of this VersionInfoDTO. Build branch

**Returns** The build\_branch of this VersionInfoDTO.

**Return type** str

**build\_revision**

Gets the build\_revision of this VersionInfoDTO. Build revision or commit hash

**Returns** The build\_revision of this VersionInfoDTO.

**Return type** str

**build\_tag**

Gets the build\_tag of this VersionInfoDTO. Build tag

**Returns** The build\_tag of this VersionInfoDTO.

**Return type** str

**build\_timestamp**

Gets the build\_timestamp of this VersionInfoDTO. Build timestamp

**Returns** The build\_timestamp of this VersionInfoDTO.

**Return type** datetime

**java\_vendor**

Gets the java\_vendor of this VersionInfoDTO. Java JVM vendor

**Returns** The java\_vendor of this VersionInfoDTO.

**Return type** str

**java\_version**

Gets the java\_version of this VersionInfoDTO. Java version

**Returns** The java\_version of this VersionInfoDTO.

**Return type** str

**ni\_fi\_version**

Gets the ni\_fi\_version of this VersionInfoDTO. The version of this NiFi.

**Returns** The ni\_fi\_version of this VersionInfoDTO.

**Return type** str

**os\_architecture**

Gets the os\_architecture of this VersionInfoDTO. Host operating system architecture

**Returns** The os\_architecture of this VersionInfoDTO.

**Return type** str

**os\_name**

Gets the os\_name of this VersionInfoDTO. Host operating system name

**Returns** The os\_name of this VersionInfoDTO.

**Return type** str

**os\_version**

Gets the os\_version of this VersionInfoDTO. Host operating system version

**Returns** The os\_version of this VersionInfoDTO.

**Return type** str

**swagger\_types** = {'build\_branch': 'str', 'build\_revision': 'str', 'build\_tag': 'str'

**to\_dict**()

Returns the model properties as a dict

**to\_str**()

Returns the string representation of the model

**nipyapi.nifi.models.versioned\_connection module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_connection.VersionedConnection (identifier=None,
                                                                    in-
                                                                    stance_identifier=None,
                                                                    name=None,
                                                                    com-
                                                                    ments=None,
                                                                    posi-
                                                                    tion=None,
                                                                    source=None,
                                                                    destina-
                                                                    tion=None,
                                                                    la-
                                                                    bel_index=None,
                                                                    z_index=None,
                                                                    se-
                                                                    lected_relationships=None,
                                                                    back_pressure_object_threshold=None,
                                                                    back_pressure_data_size_threshold=None,
                                                                    flow_file_expiration=None,
                                                                    prioritiz-
                                                                    ers=None,
                                                                    bends=None,
                                                                    load_balance_strategy=None,
                                                                    partition-
                                                                    ing_attribute=None,
                                                                    load_balance_compression=None,
                                                                    compo-
                                                                    nent_type=None,
                                                                    group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedConnection - a model defined in Swagger

```
attribute_map = {'back_pressure_data_size_threshold':  'backPressureDataSizeThreshold'}
```

#### **back\_pressure\_data\_size\_threshold**

Gets the back\_pressure\_data\_size\_threshold of this VersionedConnection. The object data size threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The back\_pressure\_data\_size\_threshold of this VersionedConnection.

**Return type** str

#### **back\_pressure\_object\_threshold**

Gets the back\_pressure\_object\_threshold of this VersionedConnection. The object count threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The back\_pressure\_object\_threshold of this VersionedConnection.

**Return type** int

#### **bends**

Gets the bends of this VersionedConnection. The bend points on the connection.

**Returns** The bends of this VersionedConnection.

**Return type** list[Position]

**comments**

Gets the comments of this VersionedConnection. The user-supplied comments for the component

**Returns** The comments of this VersionedConnection.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedConnection.

**Returns** The component\_type of this VersionedConnection.

**Return type** str

**destination**

Gets the destination of this VersionedConnection. The destination of the connection.

**Returns** The destination of this VersionedConnection.

**Return type** *ConnectableComponent*

**flow\_file\_expiration**

Gets the flow\_file\_expiration of this VersionedConnection. The amount of time a flow file may be in the flow before it will be automatically aged out of the flow. Once a flow file reaches this age it will be terminated from the flow the next time a processor attempts to start work on it.

**Returns** The flow\_file\_expiration of this VersionedConnection.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedConnection. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedConnection.

**Return type** str

**identifier**

Gets the identifier of this VersionedConnection. The component's unique identifier

**Returns** The identifier of this VersionedConnection.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedConnection. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedConnection.

**Return type** str

**label\_index**

Gets the label\_index of this VersionedConnection. The index of the bend point where to place the connection label.

**Returns** The label\_index of this VersionedConnection.

**Return type** int

**load\_balance\_compression**

Gets the `load_balance_compression` of this `VersionedConnection`. Whether or not compression should be used when transferring FlowFiles between nodes

**Returns** The `load_balance_compression` of this `VersionedConnection`.

**Return type** `str`

**load\_balance\_strategy**

Gets the `load_balance_strategy` of this `VersionedConnection`. The Strategy to use for load balancing data across the cluster, or null, if no Load Balance Strategy has been specified.

**Returns** The `load_balance_strategy` of this `VersionedConnection`.

**Return type** `str`

**name**

Gets the name of this `VersionedConnection`. The component's name

**Returns** The name of this `VersionedConnection`.

**Return type** `str`

**partitioning\_attribute**

Gets the `partitioning_attribute` of this `VersionedConnection`. The attribute to use for partitioning data as it is load balanced across the cluster. If the Load Balance Strategy is configured to use `PARTITION_BY_ATTRIBUTE`, the value returned by this method is the name of the FlowFile Attribute that will be used to determine which node in the cluster should receive a given FlowFile. If the Load Balance Strategy is unset or is set to any other value, the Partitioning Attribute has no effect.

**Returns** The `partitioning_attribute` of this `VersionedConnection`.

**Return type** `str`

**position**

Gets the position of this `VersionedConnection`. The component's position on the graph

**Returns** The position of this `VersionedConnection`.

**Return type** `Position`

**prioritizers**

Gets the prioritizers of this `VersionedConnection`. The comparators used to prioritize the queue.

**Returns** The prioritizers of this `VersionedConnection`.

**Return type** `list[str]`

**selected\_relationships**

Gets the `selected_relationships` of this `VersionedConnection`. The selected relationship that comprise the connection.

**Returns** The `selected_relationships` of this `VersionedConnection`.

**Return type** `list[str]`

**source**

Gets the source of this `VersionedConnection`. The source of the connection.

**Returns** The source of this `VersionedConnection`.

**Return type** `ConnectableComponent`

**swagger\_types** = {'back\_pressure\_data\_size\_threshold': 'str', 'back\_pressure\_object\_th

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**z\_index**

Gets the z\_index of this VersionedConnection. The z index of the connection.

**Returns** The z\_index of this VersionedConnection.

**Return type** int

## nipyapi.nifi.models.versioned\_controller\_service module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_controller\_service.**VersionedControllerService** (identifier=None)

in-  
stance\_id  
name=None  
com-  
ments=None  
po-  
si-  
tion=None,  
type=None,  
bun-  
dle=None,  
prop-  
er-  
ties=None,  
prop-  
erty\_descrip-  
con-  
troller\_servi-  
an-  
no-  
ta-  
tion\_data=None  
sched-  
uled\_state=None  
bul-  
letin\_level=None  
com-  
po-  
nent\_type=None  
group\_identi-

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedControllerService - a model defined in Swagger

**annotation\_data**

Gets the annotation\_data of this VersionedControllerService. The annotation for the controller service. This is how the custom UI relays configuration to the controller service.

**Returns** The annotation\_data of this VersionedControllerService.

**Return type** str

**attribute\_map** = {'annotation\_data': 'annotationData', 'bulletin\_level': 'bulletinLev

**bulletin\_level**

Gets the bulletin\_level of this VersionedControllerService. The level at which the controller service will report bulletins.

**Returns** The bulletin\_level of this VersionedControllerService.

**Return type** str

**bundle**

Gets the bundle of this VersionedControllerService. Information about the bundle from which the component came

**Returns** The bundle of this VersionedControllerService.

**Return type** *Bundle*

**comments**

Gets the comments of this VersionedControllerService. The user-supplied comments for the component

**Returns** The comments of this VersionedControllerService.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedControllerService.

**Returns** The component\_type of this VersionedControllerService.

**Return type** str

**controller\_service\_apis**

Gets the controller\_service\_apis of this VersionedControllerService. Lists the APIs this Controller Service implements.

**Returns** The controller\_service\_apis of this VersionedControllerService.

**Return type** list[*ControllerServiceAPI*]

**group\_identifier**

Gets the group\_identifier of this VersionedControllerService. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedControllerService.

**Return type** str

**identifier**

Gets the identifier of this VersionedControllerService. The component's unique identifier

**Returns** The identifier of this VersionedControllerService.

**Return type** str



**instance\_identifier**

Gets the instance\_identifier of this VersionedControllerService. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedControllerService.

**Return type** str

**name**

Gets the name of this VersionedControllerService. The component's name

**Returns** The name of this VersionedControllerService.

**Return type** str

**position**

Gets the position of this VersionedControllerService. The component's position on the graph

**Returns** The position of this VersionedControllerService.

**Return type** Position

**properties**

Gets the properties of this VersionedControllerService. The properties for the component. Properties whose value is not set will only contain the property name.

**Returns** The properties of this VersionedControllerService.

**Return type** dict(str, str)

**property\_descriptors**

Gets the property\_descriptors of this VersionedControllerService. The property descriptors for the component.

**Returns** The property\_descriptors of this VersionedControllerService.

**Return type** dict(str, *VersionedPropertyDescriptor*)

**scheduled\_state**

Gets the scheduled\_state of this VersionedControllerService. The ScheduledState denoting whether the Controller Service is ENABLED or DISABLED

**Returns** The scheduled\_state of this VersionedControllerService.

**Return type** str

**swagger\_types** = {'annotation\_data': 'str', 'bulletin\_level': 'str', 'bundle': 'Bund

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this VersionedControllerService. The type of the extension component

**Returns** The type of this VersionedControllerService.

**Return type** str

**nipyapi.nifi.models.versioned\_flow module**

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow.VersionedFlow (link=None, identifier=None, name=None, description=None, bucket_identifier=None, bucket_name=None, created_timestamp=None, modified_timestamp=None, type=None, permissions=None, version_count=None, revision=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlow - a model defined in Swagger

```
attribute_map = {'bucket_identifier': 'bucketIdentifier', 'bucket_name': 'bucketName'}
```

#### **bucket\_identifier**

Gets the bucket\_identifier of this VersionedFlow. The identifier of the bucket this items belongs to. This cannot be changed after the item is created.

**Returns** The bucket\_identifier of this VersionedFlow.

**Return type** str

#### **bucket\_name**

Gets the bucket\_name of this VersionedFlow. The name of the bucket this items belongs to.

**Returns** The bucket\_name of this VersionedFlow.

**Return type** str

#### **created\_timestamp**

Gets the created\_timestamp of this VersionedFlow. The timestamp of when the item was created, as milliseconds since epoch.

**Returns** The created\_timestamp of this VersionedFlow.

**Return type** int

#### **description**

Gets the description of this VersionedFlow. A description of the item.

**Returns** The description of this VersionedFlow.

**Return type** str

#### **identifier**

Gets the identifier of this VersionedFlow. An ID to uniquely identify this object.

**Returns** The identifier of this VersionedFlow.

**Return type** str

#### **link**

Gets the link of this VersionedFlow. An WebLink to this entity.

**Returns** The link of this VersionedFlow.

**Return type** JaxbLink

#### **modified\_timestamp**

Gets the modified\_timestamp of this VersionedFlow. The timestamp of when the item was last modified, as milliseconds since epoch.

**Returns** The modified\_timestamp of this VersionedFlow.

**Return type** int

#### **name**

Gets the name of this VersionedFlow. The name of the item.

**Returns** The name of this VersionedFlow.

**Return type** str

#### **permissions**

Gets the permissions of this VersionedFlow. The access that the current user has to the bucket containing this item.

**Returns** The permissions of this VersionedFlow.

**Return type** *Permissions*

#### **revision**

Gets the revision of this VersionedFlow. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this VersionedFlow.

**Return type** RevisionInfo

**swagger\_types** = {'bucket\_identifier': 'str', 'bucket\_name': 'str', 'created\_timestamp': 'str', 'name': 'str', 'permissions': 'Permissions', 'revision': 'int', 'type': 'str', 'version\_count': 'int', 'version\_id': 'str'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **type**

Gets the type of this VersionedFlow. The type of item.

**Returns** The type of this VersionedFlow.

**Return type** str

#### **version\_count**

Gets the version\_count of this VersionedFlow. The number of versions of this flow.

**Returns** The version\_count of this VersionedFlow.

**Return type** int

## **nipyapi.nifi.models.versioned\_flow\_coordinates module**

### **NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow_coordinates.VersionedFlowCoordinates (registry_url=None, bucket_id=None, flow_id=None, version=None, latest=None, registry_url=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowCoordinates - a model defined in Swagger

```
attribute_map = {'bucket_id': 'bucketId', 'flow_id': 'flowId', 'latest': 'latest',
```

```
bucket_id
```

Gets the bucket\_id of this VersionedFlowCoordinates. The UUID of the bucket that the flow resides in

**Returns** The bucket\_id of this VersionedFlowCoordinates.

**Return type** str

```
flow_id
```

Gets the flow\_id of this VersionedFlowCoordinates. The UUID of the flow

**Returns** The flow\_id of this VersionedFlowCoordinates.

**Return type** str

```
latest
```

Gets the latest of this VersionedFlowCoordinates. Whether or not these coordinates point to the latest version of the flow

**Returns** The latest of this VersionedFlowCoordinates.

**Return type** bool

```
registry_url
```

Gets the registry\_url of this VersionedFlowCoordinates. The URL of the Flow Registry that contains the flow

**Returns** The registry\_url of this VersionedFlowCoordinates.

**Return type** str

```
swagger_types = {'bucket_id': 'str', 'flow_id': 'str', 'latest': 'bool', 'registry_url': 'str'}
```

```
to_dict()
```

Returns the model properties as a dict

```
to_str()
```

Returns the string representation of the model

```
version
```

Gets the version of this VersionedFlowCoordinates. The version of the flow

**Returns** The version of this VersionedFlowCoordinates.

**Return type** int

## nipyapi.nifi.models.versioned\_flow\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO(registry_id=None,
                                                             bucket_id=None,
                                                             flow_id=None,
                                                             flow_name=None,
                                                             descrip-
                                                             tion=None,
                                                             comments=None,
                                                             action=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowDTO - a model defined in Swagger

#### **action**

Gets the action of this VersionedFlowDTO. The action being performed

**Returns** The action of this VersionedFlowDTO.

**Return type** str

```
attribute_map = {'action':  'action', 'bucket_id':  'bucketId', 'comments':  'comments'}
```

#### **bucket\_id**

Gets the bucket\_id of this VersionedFlowDTO. The ID of the bucket where the flow is stored

**Returns** The bucket\_id of this VersionedFlowDTO.

**Return type** str

#### **comments**

Gets the comments of this VersionedFlowDTO. Comments for the changeset

**Returns** The comments of this VersionedFlowDTO.

**Return type** str

#### **description**

Gets the description of this VersionedFlowDTO. A description of the flow

**Returns** The description of this VersionedFlowDTO.

**Return type** str

#### **flow\_id**

Gets the flow\_id of this VersionedFlowDTO. The ID of the flow

**Returns** The flow\_id of this VersionedFlowDTO.

**Return type** str

#### **flow\_name**

Gets the flow\_name of this VersionedFlowDTO. The name of the flow

**Returns** The flow\_name of this VersionedFlowDTO.

**Return type** str

**registry\_id**

Gets the registry\_id of this VersionedFlowDTO. The ID of the registry that the flow is tracked to

**Returns** The registry\_id of this VersionedFlowDTO.

**Return type** str

**swagger\_types** = {'action': 'str', 'bucket\_id': 'str', 'comments': 'str', 'descripti

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.nifi.models.versioned\_flow\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_flow\_entity.**VersionedFlowEntity**(versioned\_flow=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowEntity - a model defined in Swagger

**attribute\_map** = {'versioned\_flow': 'versionedFlow'}

**swagger\_types** = {'versioned\_flow': 'VersionedFlowDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_flow**

Gets the versioned\_flow of this VersionedFlowEntity. The versioned flow

**Returns** The versioned\_flow of this VersionedFlowEntity.

**Return type** *VersionedFlowDTO*

## nipyapi.nifi.models.versioned\_flow\_snapshot module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow_snapshot.VersionedFlowSnapshot (snapshot_metadata=None,
                                                                    flow_contents=None,
                                                                    ex-
                                                                    ter-
                                                                    nal_controller_services=None,
                                                                    pa-
                                                                    ram-
                                                                    e-
                                                                    ter_contexts=None,
                                                                    flow_encoding_version=None,
                                                                    flow=None,
                                                                    bucket=None,
                                                                    lat-
                                                                    est=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshot - a model defined in Swagger

```
attribute_map = {'bucket': 'bucket', 'external_controller_services': 'externalContro
```

#### **bucket**

Gets the bucket of this VersionedFlowSnapshot. The bucket where the flow is located

**Returns** The bucket of this VersionedFlowSnapshot.

**Return type** *Bucket*

#### **external\_controller\_services**

Gets the external\_controller\_services of this VersionedFlowSnapshot. The information about controller services that exist outside this versioned flow, but are referenced by components within the versioned flow.

**Returns** The external\_controller\_services of this VersionedFlowSnapshot.

**Return type** dict(str, ExternalControllerServiceReference)

#### **flow**

Gets the flow of this VersionedFlowSnapshot. The flow this snapshot is for

**Returns** The flow of this VersionedFlowSnapshot.

**Return type** *VersionedFlow*

#### **flow\_contents**

Gets the flow\_contents of this VersionedFlowSnapshot. The contents of the versioned flow

**Returns** The flow\_contents of this VersionedFlowSnapshot.

**Return type** *VersionedProcessGroup*

#### **flow\_encoding\_version**

Gets the flow\_encoding\_version of this VersionedFlowSnapshot. The optional encoding version of the flow contents.

**Returns** The flow\_encoding\_version of this VersionedFlowSnapshot.

**Return type** str

#### **latest**

Gets the latest of this VersionedFlowSnapshot.

**Returns** The latest of this VersionedFlowSnapshot.

**Return type** bool

**parameter\_contexts**

Gets the parameter\_contexts of this VersionedFlowSnapshot. The parameter contexts referenced by process groups in the flow contents. The mapping is from the name of the context to the context instance, and it is expected that any context in this map is referenced by at least one process group in this flow.

**Returns** The parameter\_contexts of this VersionedFlowSnapshot.

**Return type** dict(str, VersionedParameterContext)

**snapshot\_metadata**

Gets the snapshot\_metadata of this VersionedFlowSnapshot. The metadata for this snapshot

**Returns** The snapshot\_metadata of this VersionedFlowSnapshot.

**Return type** *VersionedFlowSnapshotMetadata*

**swagger\_types** = {'bucket': 'Bucket', 'external\_controller\_services': 'dict(str, Ext

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.nifi.models.versioned\_flow\_snapshot\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity (versioned_flow_snapshot_entity)
    process_group_id: str
    registry_id: str
    update_date: datetime
    disconnected_node_acknowledged: bool
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshotEntity - a model defined in Swagger

**attribute\_map** = {'disconnected\_node\_acknowledged': 'disconnectedNodeAcknowledged', 'p

**disconnected\_node\_acknowledged**

Gets the disconnected\_node\_acknowledged of this VersionedFlowSnapshotEntity. Acknowledges that this node is disconnected to allow for mutable requests to proceed.

**Returns** The disconnected\_node\_acknowledged of this VersionedFlowSnapshotEntity.



**Return type** bool

**process\_group\_revision**

Gets the process\_group\_revision of this VersionedFlowSnapshotEntity. The Revision of the Process Group under Version Control

**Returns** The process\_group\_revision of this VersionedFlowSnapshotEntity.

**Return type** *RevisionDTO*

**registry\_id**

Gets the registry\_id of this VersionedFlowSnapshotEntity. The ID of the Registry that this flow belongs to

**Returns** The registry\_id of this VersionedFlowSnapshotEntity.

**Return type** str

**swagger\_types** = {'disconnected\_node\_acknowledged': 'bool', 'process\_group\_revision':

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**update\_descendant\_versioned\_flows**

Gets the update\_descendant\_versioned\_flows of this VersionedFlowSnapshotEntity. If the Process Group to be updated has a child or descendant Process Group that is also under Version Control, this specifies whether or not the contents of that child/descendant Process Group should be updated.

**Returns** The update\_descendant\_versioned\_flows of this VersionedFlowSnapshotEntity.

**Return type** bool

**versioned\_flow\_snapshot**

Gets the versioned\_flow\_snapshot of this VersionedFlowSnapshotEntity. The versioned flow snapshot

**Returns** The versioned\_flow\_snapshot of this VersionedFlowSnapshotEntity.

**Return type** *VersionedFlowSnapshot*

## nipyapi.nifi.models.versioned\_flow\_snapshot\_metadata\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_flow\_snapshot\_metadata\_entity.**VersionedFlowSnapshotMeta**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshotMetadataEntity - a model defined in Swagger

**attribute\_map** = {'registry\_id': 'registryId', 'versioned\_flow\_snapshot\_metadata': 'v

**registry\_id**

Gets the registry\_id of this VersionedFlowSnapshotMetadataEntity. The ID of the Registry that this flow belongs to

**Returns** The registry\_id of this VersionedFlowSnapshotMetadataEntity.

**Return type** str

**swagger\_types** = {'registry\_id': 'str', 'versioned\_flow\_snapshot\_metadata': 'VersionedFlowSnapshotMetadataEntity'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_flow\_snapshot\_metadata**

Gets the versioned\_flow\_snapshot\_metadata of this VersionedFlowSnapshotMetadataEntity. The collection of versioned flow snapshot metadata

**Returns** The versioned\_flow\_snapshot\_metadata of this VersionedFlowSnapshotMetadataEntity.

**Return type** *VersionedFlowSnapshotMetadata*

## nipyapi.nifi.models.versioned\_flow\_snapshot\_metadata\_set\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_flow\_snapshot\_metadata\_set\_entity.**VersionedFlowSnapshotMetadataSetEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshotMetadataSetEntity - a model defined in Swagger

**attribute\_map** = {'versioned\_flow\_snapshot\_metadata\_set': 'versionedFlowSnapshotMetadataSet'}

**swagger\_types** = {'versioned\_flow\_snapshot\_metadata\_set': 'list[VersionedFlowSnapshotMetadataEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_flow\_snapshot\_metadata\_set**

Gets the versioned\_flow\_snapshot\_metadata\_set of this VersionedFlowSnapshotMetadataSetEntity.

**Returns** The versioned\_flow\_snapshot\_metadata\_set of this VersionedFlowSnapshotMetadataSetEntity.

**Return type** list[*VersionedFlowSnapshotMetadataEntity*]

## nipyapi.nifi.models.versioned\_flow\_update\_request\_dto module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_flow_update_request_dto.VersionedFlowUpdateRequestDTO (
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowUpdateRequestDTO - a model defined in Swagger

```
attribute_map = {'complete': 'complete', 'failure_reason': 'failureReason', 'last_up
```

**complete**

Gets the complete of this VersionedFlowUpdateRequestDTO. Whether or not this request has completed

**Returns** The complete of this VersionedFlowUpdateRequestDTO.

**Return type** bool

**failure\_reason**

Gets the failure\_reason of this VersionedFlowUpdateRequestDTO. An explanation of why this request failed, or null if this request has not failed

**Returns** The failure\_reason of this VersionedFlowUpdateRequestDTO.

**Return type** str

**last\_updated**

Gets the last\_updated of this VersionedFlowUpdateRequestDTO. The last time this request was updated.

**Returns** The last\_updated of this VersionedFlowUpdateRequestDTO.

**Return type** str

**percent\_completed**

Gets the percent\_completed of this VersionedFlowUpdateRequestDTO. The percentage complete for the request, between 0 and 100

**Returns** The percent\_completed of this VersionedFlowUpdateRequestDTO.

**Return type** int

**process\_group\_id**

Gets the process\_group\_id of this VersionedFlowUpdateRequestDTO. The unique ID of the Process Group being updated

**Returns** The process\_group\_id of this VersionedFlowUpdateRequestDTO.

**Return type** str

**request\_id**

Gets the request\_id of this VersionedFlowUpdateRequestDTO. The unique ID of this request.

**Returns** The request\_id of this VersionedFlowUpdateRequestDTO.

**Return type** str

**state**

Gets the state of this VersionedFlowUpdateRequestDTO. The state of the request

**Returns** The state of this VersionedFlowUpdateRequestDTO.

**Return type** str

**swagger\_types** = {'complete': 'bool', 'failure\_reason': 'str', 'last\_updated': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**uri**

Gets the uri of this VersionedFlowUpdateRequestDTO. The URI for future requests to this drop request.

**Returns** The uri of this VersionedFlowUpdateRequestDTO.

**Return type** str

**version\_control\_information**

Gets the version\_control\_information of this VersionedFlowUpdateRequestDTO. The VersionControlInformation that describes where the Versioned Flow is located; this may not be populated until the request is completed.

**Returns** The version\_control\_information of this VersionedFlowUpdateRequestDTO.

**Return type** *VersionControlInformationDTO*

## nipyapi.nifi.models.versioned\_flow\_update\_request\_entity module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_flow\_update\_request\_entity.**VersionedFlowUpdateRequestEntity**

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowUpdateRequestEntity - a model defined in Swagger

**attribute\_map** = {'process\_group\_revision': 'processGroupRevision', 'request': 'request'}

**process\_group\_revision**

Gets the process\_group\_revision of this VersionedFlowUpdateRequestEntity. The revision for the Process Group being updated.

**Returns** The process\_group\_revision of this VersionedFlowUpdateRequestEntity.

**Return type** *RevisionDTO*

**request**

Gets the request of this VersionedFlowUpdateRequestEntity. The Flow Update Request

**Returns** The request of this VersionedFlowUpdateRequestEntity.

**Return type** *VersionedFlowUpdateRequestDTO*

**swagger\_types** = {'process\_group\_revision': 'RevisionDTO', 'request': 'VersionedFlowUpdateRequestDTO'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

### nipyapi.nifi.models.versioned\_flows\_entity module

NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.nifi.models.versioned\_flows\_entity.**VersionedFlowsEntity**(versioned\_flows=None)  
Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowsEntity - a model defined in Swagger

**attribute\_map** = {'versioned\_flows': 'versionedFlows'}

**swagger\_types** = {'versioned\_flows': 'list[VersionedFlowEntity]'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**versioned\_flows**

Gets the versioned\_flows of this VersionedFlowsEntity.

**Returns** The versioned\_flows of this VersionedFlowsEntity.

**Return type** list[*VersionedFlowEntity*]

## nipyapi.nifi.models.versioned\_funnel module

### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_funnel.VersionedFunnel (identifier=None, instance_identifier=None, name=None, comments=None, position=None, component_type=None, group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFunnel - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'component_type': 'componentType', 'group_i
```

#### **comments**

Gets the comments of this VersionedFunnel. The user-supplied comments for the component

**Returns** The comments of this VersionedFunnel.

**Return type** str

#### **component\_type**

Gets the component\_type of this VersionedFunnel.

**Returns** The component\_type of this VersionedFunnel.

**Return type** str

#### **group\_identifier**

Gets the group\_identifier of this VersionedFunnel. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedFunnel.

**Return type** str

#### **identifier**

Gets the identifier of this VersionedFunnel. The component's unique identifier

**Returns** The identifier of this VersionedFunnel.

**Return type** str

#### **instance\_identifier**

Gets the instance\_identifier of this VersionedFunnel. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedFunnel.

**Return type** str

#### **name**

Gets the name of this VersionedFunnel. The component's name

**Returns** The name of this VersionedFunnel.

**Return type** str

**position**

Gets the position of this VersionedFunnel. The component's position on the graph

**Returns** The position of this VersionedFunnel.

**Return type** Position

**swagger\_types** = {'comments': 'str', 'component\_type': 'str', 'group\_identifier': 's

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

### nipyapi.nifi.models.versioned\_label module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_label.VersionedLabel (identifier=None,
                                                         instance_identifier=None,
                                                         name=None,
                                                         comments=None,
                                                         position=None,
                                                         label=None,
                                                         z_index=None,
                                                         width=None,
                                                         height=None,
                                                         style=None,
                                                         component_type=None,
                                                         group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedLabel - a model defined in Swagger

**attribute\_map** = {'comments': 'comments', 'component\_type': 'componentType', 'group\_i

**comments**

Gets the comments of this VersionedLabel. The user-supplied comments for the component

**Returns** The comments of this VersionedLabel.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedLabel.

**Returns** The component\_type of this VersionedLabel.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedLabel. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedLabel.

**Return type** str

**height**

Gets the height of this VersionedLabel. The height of the label in pixels when at a 1:1 scale.

**Returns** The height of this VersionedLabel.

**Return type** float

**identifier**

Gets the identifier of this VersionedLabel. The component's unique identifier

**Returns** The identifier of this VersionedLabel.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedLabel. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedLabel.

**Return type** str

**label**

Gets the label of this VersionedLabel. The text that appears in the label.

**Returns** The label of this VersionedLabel.

**Return type** str

**name**

Gets the name of this VersionedLabel. The component's name

**Returns** The name of this VersionedLabel.

**Return type** str

**position**

Gets the position of this VersionedLabel. The component's position on the graph

**Returns** The position of this VersionedLabel.

**Return type** Position

**style**

Gets the style of this VersionedLabel. The styles for this label (font-size : 12px, background-color : #eee, etc).

**Returns** The style of this VersionedLabel.

**Return type** dict(str, str)

**swagger\_types** = {'comments': 'str', 'component\_type': 'str', 'group\_identifier': 's

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model



**width**

Gets the width of this VersionedLabel. The width of the label in pixels when at a 1:1 scale.

**Returns** The width of this VersionedLabel.

**Return type** float

**z\_index**

Gets the z\_index of this VersionedLabel. The z index of the connection.

**Returns** The z\_index of this VersionedLabel.

**Return type** int

**nipyapi.nifi.models.versioned\_port module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_port.VersionedPort (identifier=None, instance_identifier=None, name=None, comments=None, position=None, type=None, concurrently_schedulable_task_count=None, scheduled_state=None, allow_remote_access=None, component_type=None, group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedPort - a model defined in Swagger

**allow\_remote\_access**

Gets the allow\_remote\_access of this VersionedPort. Whether or not this port allows remote access for site-to-site

**Returns** The allow\_remote\_access of this VersionedPort.

**Return type** bool

```
attribute_map = {'allow_remote_access': 'allowRemoteAccess', 'comments': 'comments',
```

**comments**

Gets the comments of this VersionedPort. The user-supplied comments for the component

**Returns** The comments of this VersionedPort.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedPort.

**Returns** The component\_type of this VersionedPort.

**Return type** str

**concurrently\_schedulable\_task\_count**

Gets the `concurrently_schedulable_task_count` of this `VersionedPort`. The number of tasks that should be concurrently scheduled for the port.

**Returns** The `concurrently_schedulable_task_count` of this `VersionedPort`.

**Return type** int

**group\_identifier**

Gets the `group_identifier` of this `VersionedPort`. The ID of the Process Group that this component belongs to

**Returns** The `group_identifier` of this `VersionedPort`.

**Return type** str

**identifier**

Gets the identifier of this `VersionedPort`. The component's unique identifier

**Returns** The identifier of this `VersionedPort`.

**Return type** str

**instance\_identifier**

Gets the `instance_identifier` of this `VersionedPort`. The instance ID of an existing component that is described by this `VersionedComponent`, or null if this is not mapped to an instantiated component

**Returns** The `instance_identifier` of this `VersionedPort`.

**Return type** str

**name**

Gets the name of this `VersionedPort`. The component's name

**Returns** The name of this `VersionedPort`.

**Return type** str

**position**

Gets the position of this `VersionedPort`. The component's position on the graph

**Returns** The position of this `VersionedPort`.

**Return type** Position

**scheduled\_state**

Gets the `scheduled_state` of this `VersionedPort`. The scheduled state of the component

**Returns** The `scheduled_state` of this `VersionedPort`.

**Return type** str

**swagger\_types** = {'allow\_remote\_access': 'bool', 'comments': 'str', 'component\_type':

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

**type**

Gets the type of this `VersionedPort`. The type of port.

**Returns** The type of this `VersionedPort`.

**Return type** str

### **nipyapi.nifi.models.versioned\_process\_group module**

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup (identifier=None,
                                                                    in-
                                                                    stance_identifier=None,
                                                                    name=None,
                                                                    com-
                                                                    ments=None,
                                                                    po-
                                                                    si-
                                                                    tion=None,
                                                                    pro-
                                                                    cess_groups=None,
                                                                    re-
                                                                    mote_process_groups=None,
                                                                    pro-
                                                                    ces-
                                                                    sors=None,
                                                                    in-
                                                                    put_ports=None,
                                                                    out-
                                                                    put_ports=None,
                                                                    con-
                                                                    nec-
                                                                    tions=None,
                                                                    la-
                                                                    bels=None,
                                                                    fun-
                                                                    nels=None,
                                                                    con-
                                                                    troller_services=None,
                                                                    ver-
                                                                    sioned_flow_coordinates=None,
                                                                    vari-
                                                                    ables=None,
                                                                    pa-
                                                                    ram-
                                                                    e-
                                                                    ter_context_name=None,
                                                                    de-
                                                                    fault_flow_file_expiration=None,
                                                                    de-
                                                                    fault_back_pressure_object=None,
                                                                    de-
                                                                    fault_back_pressure_data=None,
                                                                    com-
                                                                    po-
                                                                    nent_type=None,
                                                                    flow_file_concurrency=None,
                                                                    flow_file_outbound_policy=None,
                                                                    group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedProcessGroup - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'component_type': 'componentType', 'connect
```

**comments**

Gets the comments of this VersionedProcessGroup. The user-supplied comments for the component

**Returns** The comments of this VersionedProcessGroup.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedProcessGroup.

**Returns** The component\_type of this VersionedProcessGroup.

**Return type** str

**connections**

Gets the connections of this VersionedProcessGroup. The Connections

**Returns** The connections of this VersionedProcessGroup.

**Return type** list[[VersionedConnection](#)]

**controller\_services**

Gets the controller\_services of this VersionedProcessGroup. The Controller Services

**Returns** The controller\_services of this VersionedProcessGroup.

**Return type** list[[VersionedControllerService](#)]

**default\_back\_pressure\_data\_size\_threshold**

Gets the default\_back\_pressure\_data\_size\_threshold of this VersionedProcessGroup. Default value used in this Process Group for the maximum data size of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_data\_size\_threshold of this VersionedProcessGroup.

**Return type** str

**default\_back\_pressure\_object\_threshold**

Gets the default\_back\_pressure\_object\_threshold of this VersionedProcessGroup. Default value used in this Process Group for the maximum number of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_object\_threshold of this VersionedProcessGroup.

**Return type** int

**default\_flow\_file\_expiration**

Gets the default\_flow\_file\_expiration of this VersionedProcessGroup. The default FlowFile Expiration for this Process Group.

**Returns** The default\_flow\_file\_expiration of this VersionedProcessGroup.

**Return type** str

**flow\_file\_concurrency**

Gets the flow\_file\_concurrency of this VersionedProcessGroup. The configured FlowFile Concurrency for the Process Group

**Returns** The flow\_file\_concurrency of this VersionedProcessGroup.

**Return type** str

**flow\_file\_outbound\_policy**

Gets the flow\_file\_outbound\_policy of this VersionedProcessGroup. The FlowFile Outbound Policy for the Process Group

**Returns** The flow\_file\_outbound\_policy of this VersionedProcessGroup.

**Return type** str

**funnels**

Gets the funnels of this VersionedProcessGroup. The Funnels

**Returns** The funnels of this VersionedProcessGroup.

**Return type** list[[VersionedFunnel](#)]

**group\_identifier**

Gets the group\_identifier of this VersionedProcessGroup. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedProcessGroup.

**Return type** str

**identifier**

Gets the identifier of this VersionedProcessGroup. The component's unique identifier

**Returns** The identifier of this VersionedProcessGroup.

**Return type** str

**input\_ports**

Gets the input\_ports of this VersionedProcessGroup. The Input Ports

**Returns** The input\_ports of this VersionedProcessGroup.

**Return type** list[[VersionedPort](#)]

**instance\_identifier**

Gets the instance\_identifier of this VersionedProcessGroup. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedProcessGroup.

**Return type** str

**labels**

Gets the labels of this VersionedProcessGroup. The Labels

**Returns** The labels of this VersionedProcessGroup.

**Return type** list[[VersionedLabel](#)]

**name**

Gets the name of this VersionedProcessGroup. The component's name

**Returns** The name of this VersionedProcessGroup.

**Return type** str

**output\_ports**

Gets the output\_ports of this VersionedProcessGroup. The Output Ports

**Returns** The output\_ports of this VersionedProcessGroup.

**Return type** list[[VersionedPort](#)]

**parameter\_context\_name**

Gets the parameter\_context\_name of this VersionedProcessGroup. The name of the parameter context used by this process group

**Returns** The parameter\_context\_name of this VersionedProcessGroup.

**Return type** str

**position**

Gets the position of this VersionedProcessGroup. The component's position on the graph

**Returns** The position of this VersionedProcessGroup.

**Return type** Position

**process\_groups**

Gets the process\_groups of this VersionedProcessGroup. The child Process Groups

**Returns** The process\_groups of this VersionedProcessGroup.

**Return type** list[*VersionedProcessGroup*]

**processors**

Gets the processors of this VersionedProcessGroup. The Processors

**Returns** The processors of this VersionedProcessGroup.

**Return type** list[*VersionedProcessor*]

**remote\_process\_groups**

Gets the remote\_process\_groups of this VersionedProcessGroup. The Remote Process Groups

**Returns** The remote\_process\_groups of this VersionedProcessGroup.

**Return type** list[*VersionedRemoteProcessGroup*]

**swagger\_types** = {'comments': 'str', 'component\_type': 'str', 'connections': 'list[V

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**variables**

Gets the variables of this VersionedProcessGroup. The Variables in the Variable Registry for this Process Group (not including any ancestor or descendant Process Groups)

**Returns** The variables of this VersionedProcessGroup.

**Return type** dict(str, str)

**versioned\_flow\_coordinates**

Gets the versioned\_flow\_coordinates of this VersionedProcessGroup. The coordinates where the remote flow is stored, or null if the Process Group is not directly under Version Control

**Returns** The versioned\_flow\_coordinates of this VersionedProcessGroup.

**Return type** *VersionedFlowCoordinates*

**nipyapi.nifi.models.versioned\_processor module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_processor.VersionedProcessor (identifier=None,  
in-  
stance_identifier=None,  
name=None,  
com-  
ments=None,  
posi-  
tion=None,  
type=None,  
bun-  
dle=None,  
proper-  
ties=None,  
prop-  
erty_descriptors=None,  
style=None,  
annota-  
tion_data=None,  
schedul-  
ing_period=None,  
schedul-  
ing_strategy=None,  
execu-  
tion_node=None,  
penalty_duration=None,  
yield_duration=None,  
bul-  
letin_level=None,  
run_duration_millis=None,  
concur-  
rently_schedulable_task_count=None,  
auto_terminated_relationships=None,  
sched-  
uled_state=None,  
retry_count=None,  
re-  
tried_relationships=None,  
back-  
off_mechanism=None,  
max_backoff_period=None,  
compo-  
nent_type=None,  
group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedProcessor - a model defined in Swagger

#### **annotation\_data**

Gets the annotation\_data of this VersionedProcessor. The annotation data for the processor used to relay configuration between a custom UI and the procesosr.

**Returns** The annotation\_data of this VersionedProcessor.

**Return type** str



```
attribute_map = {'annotation_data': 'annotationData', 'auto_terminated_relationships'
```

#### **auto\_terminated\_relationships**

Gets the auto\_terminated\_relationships of this VersionedProcessor. The names of all relationships that cause a flow file to be terminated if the relationship is not connected elsewhere. This property differs from the 'isAutoTerminate' property of the RelationshipDTO in that the RelationshipDTO is meant to depict the current configuration, whereas this property can be set in a DTO when updating a Processor in order to change which Relationships should be auto-terminated.

**Returns** The auto\_terminated\_relationships of this VersionedProcessor.

**Return type** list[str]

#### **backoff\_mechanism**

Gets the backoff\_mechanism of this VersionedProcessor. Determines whether the FlowFile should be penalized or the processor should be yielded between retries.

**Returns** The backoff\_mechanism of this VersionedProcessor.

**Return type** str

#### **bulletin\_level**

Gets the bulletin\_level of this VersionedProcessor. The level at which the processor will report bulletins.

**Returns** The bulletin\_level of this VersionedProcessor.

**Return type** str

#### **bundle**

Gets the bundle of this VersionedProcessor. Information about the bundle from which the component came

**Returns** The bundle of this VersionedProcessor.

**Return type** *Bundle*

#### **comments**

Gets the comments of this VersionedProcessor. The user-supplied comments for the component

**Returns** The comments of this VersionedProcessor.

**Return type** str

#### **component\_type**

Gets the component\_type of this VersionedProcessor.

**Returns** The component\_type of this VersionedProcessor.

**Return type** str

#### **concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this VersionedProcessor. The number of tasks that should be concurrently schedule for the processor. If the processor doesn't allow parallel processing then any positive input will be ignored.

**Returns** The concurrently\_schedulable\_task\_count of this VersionedProcessor.

**Return type** int

#### **execution\_node**

Gets the execution\_node of this VersionedProcessor. Indicates the node where the process will execute.

**Returns** The execution\_node of this VersionedProcessor.

**Return type** str

**group\_identifier**

Gets the `group_identifier` of this `VersionedProcessor`. The ID of the Process Group that this component belongs to

**Returns** The `group_identifier` of this `VersionedProcessor`.

**Return type** `str`

**identifier**

Gets the `identifier` of this `VersionedProcessor`. The component's unique identifier

**Returns** The `identifier` of this `VersionedProcessor`.

**Return type** `str`

**instance\_identifier**

Gets the `instance_identifier` of this `VersionedProcessor`. The instance ID of an existing component that is described by this `VersionedComponent`, or null if this is not mapped to an instantiated component

**Returns** The `instance_identifier` of this `VersionedProcessor`.

**Return type** `str`

**max\_backoff\_period**

Gets the `max_backoff_period` of this `VersionedProcessor`. Maximum amount of time to be waited during a retry period.

**Returns** The `max_backoff_period` of this `VersionedProcessor`.

**Return type** `str`

**name**

Gets the `name` of this `VersionedProcessor`. The component's name

**Returns** The `name` of this `VersionedProcessor`.

**Return type** `str`

**penalty\_duration**

Gets the `penalty_duration` of this `VersionedProcessor`. The amount of time that is used when the process penalizes a flowfile.

**Returns** The `penalty_duration` of this `VersionedProcessor`.

**Return type** `str`

**position**

Gets the `position` of this `VersionedProcessor`. The component's position on the graph

**Returns** The `position` of this `VersionedProcessor`.

**Return type** `Position`

**properties**

Gets the `properties` of this `VersionedProcessor`. The properties for the component. Properties whose value is not set will only contain the property name.

**Returns** The `properties` of this `VersionedProcessor`.

**Return type** `dict(str, str)`

**property\_descriptors**

Gets the `property_descriptors` of this `VersionedProcessor`. The property descriptors for the component.

**Returns** The `property_descriptors` of this `VersionedProcessor`.

**Return type** `dict(str, VersionedPropertyDescriptor)`

**retried\_relationships**

Gets the retried\_relationships of this VersionedProcessor. All the relationships should be retried.

**Returns** The retried\_relationships of this VersionedProcessor.

**Return type** list[str]

**retry\_count**

Gets the retry\_count of this VersionedProcessor. Overall number of retries.

**Returns** The retry\_count of this VersionedProcessor.

**Return type** int

**run\_duration\_millis**

Gets the run\_duration\_millis of this VersionedProcessor. The run duration for the processor in milliseconds.

**Returns** The run\_duration\_millis of this VersionedProcessor.

**Return type** int

**scheduled\_state**

Gets the scheduled\_state of this VersionedProcessor. The scheduled state of the component

**Returns** The scheduled\_state of this VersionedProcessor.

**Return type** str

**scheduling\_period**

Gets the scheduling\_period of this VersionedProcessor. The frequency with which to schedule the processor. The format of the value will depend on the value of schedulingStrategy.

**Returns** The scheduling\_period of this VersionedProcessor.

**Return type** str

**scheduling\_strategy**

Gets the scheduling\_strategy of this VersionedProcessor. Indicates whether the processor should be scheduled to run in event or timer driven mode.

**Returns** The scheduling\_strategy of this VersionedProcessor.

**Return type** str

**style**

Gets the style of this VersionedProcessor. Stylistic data for rendering in a UI

**Returns** The style of this VersionedProcessor.

**Return type** dict(str, str)

**swagger\_types** = {'annotation\_data': 'str', 'auto\_terminated\_relationships': 'list[str]

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this VersionedProcessor. The type of the extension component

**Returns** The type of this VersionedProcessor.

**Return type** str

**yield\_duration**

Gets the yield\_duration of this VersionedProcessor. The amount of time that must elapse before this processor is scheduled again after yielding.

**Returns** The yield\_duration of this VersionedProcessor.

**Return type** str

**nipyapi.nifi.models.versioned\_property\_descriptor module**

## NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_property_descriptor.VersionedPropertyDescriptor (name=No
dis-
play_name
iden-
ti-
fies_contr
sen-
si-
tive=Non
re-
source_d
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedPropertyDescriptor - a model defined in Swagger

```
attribute_map = {'display_name': 'displayName', 'identifies_controller_service': 'id
```

**display\_name**

Gets the display\_name of this VersionedPropertyDescriptor. The display name of the property

**Returns** The display\_name of this VersionedPropertyDescriptor.

**Return type** str

**identifies\_controller\_service**

Gets the identifies\_controller\_service of this VersionedPropertyDescriptor. Whether or not the property provides the identifier of a Controller Service

**Returns** The identifies\_controller\_service of this VersionedPropertyDescriptor.

**Return type** bool

**name**

Gets the name of this VersionedPropertyDescriptor. The name of the property

**Returns** The name of this VersionedPropertyDescriptor.

**Return type** str

**resource\_definition**

Gets the resource\_definition of this VersionedPropertyDescriptor. Returns the Resource Definition that defines which type(s) of resource(s) this property references, if any

**Returns** The resource\_definition of this VersionedPropertyDescriptor.

**Return type** VersionedResourceDefinition

**sensitive**

Gets the sensitive of this VersionedPropertyDescriptor. Whether or not the property is considered sensitive

**Returns** The sensitive of this VersionedPropertyDescriptor.

**Return type** bool

**swagger\_types** = {'display\_name': 'str', 'identifies\_controller\_service': 'bool', 'name': 'str', 'parent\_name': 'str', 'parent\_type': 'str', 'resource\_definition': 'VersionedResourceDefinition', 'sensitive': 'bool', 'type': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**nipyapi.nifi.models.versioned\_remote\_group\_port module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_remote_group_port.VersionedRemoteGroupPort (identifier=None,
in-
stance_identifier=None,
name=None,
com-
ments=None,
po-
si-
tion=None,
re-
mote_group_id=None,
con-
cur-
rently_scheduled=None,
use_compression=None,
batch_size=None,
com-
po-
nent_type=None,
tar-
get_id=None,
sched-
uled_state=None,
group_identifier=None)
    """
    """
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedRemoteGroupPort - a model defined in Swagger

**attribute\_map** = {'batch\_size': 'batchSize', 'comments': 'comments', 'component\_type'

**batch\_size**

Gets the batch\_size of this VersionedRemoteGroupPort. The batch settings for data transmission.

**Returns** The batch\_size of this VersionedRemoteGroupPort.

**Return type** *BatchSize*

**comments**

Gets the comments of this VersionedRemoteGroupPort. The user-supplied comments for the component

**Returns** The comments of this VersionedRemoteGroupPort.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedRemoteGroupPort.

**Returns** The component\_type of this VersionedRemoteGroupPort.

**Return type** str

**concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this VersionedRemoteGroupPort. The number of task that may transmit flowfiles to the target port concurrently.

**Returns** The concurrently\_schedulable\_task\_count of this VersionedRemoteGroupPort.

**Return type** int

**group\_identifier**

Gets the group\_identifier of this VersionedRemoteGroupPort. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedRemoteGroupPort.

**Return type** str

**identifier**

Gets the identifier of this VersionedRemoteGroupPort. The component's unique identifier

**Returns** The identifier of this VersionedRemoteGroupPort.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedRemoteGroupPort. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedRemoteGroupPort.

**Return type** str

**name**

Gets the name of this VersionedRemoteGroupPort. The component's name

**Returns** The name of this VersionedRemoteGroupPort.

**Return type** str

**position**

Gets the position of this VersionedRemoteGroupPort. The component's position on the graph

**Returns** The position of this VersionedRemoteGroupPort.

**Return type** Position

#### **remote\_group\_id**

Gets the remote\_group\_id of this VersionedRemoteGroupPort. The id of the remote process group that the port resides in.

**Returns** The remote\_group\_id of this VersionedRemoteGroupPort.

**Return type** str

#### **scheduled\_state**

Gets the scheduled\_state of this VersionedRemoteGroupPort. The scheduled state of the component

**Returns** The scheduled\_state of this VersionedRemoteGroupPort.

**Return type** str

**swagger\_types** = {'batch\_size': 'BatchSize', 'comments': 'str', 'component\_type': 's

#### **target\_id**

Gets the target\_id of this VersionedRemoteGroupPort. The ID of the port on the target NiFi instance

**Returns** The target\_id of this VersionedRemoteGroupPort.

**Return type** str

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **use\_compression**

Gets the use\_compression of this VersionedRemoteGroupPort. Whether the flowfiles are compressed when sent to the target port.

**Returns** The use\_compression of this VersionedRemoteGroupPort.

**Return type** bool

### **nipyapi.nifi.models.versioned\_remote\_process\_group module**

#### **NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.models.versioned_remote_process_group.VersionedRemoteProcessGroup (identified by its instance_id, name=NifiVersionedRemoteProcessGroupConstants.DEFAULT_NAME, comments=None, position=None, target_uri=None, target_username=None, target_password=None, target_protocol=None, target_host=None, target_port=None, target_hostname=None, proxy_host=None, proxy_port=None, proxy_username=None, proxy_password=None, proxy_protocol=None, proxy_hostname=None, put_port=None, out_port=None, put_port_name=None, component_type=None, group_id=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedRemoteProcessGroup - a model defined in Swagger

**attribute\_map** = {'comments': 'comments', 'communications\_timeout': 'communicationsTimeout'}

#### **comments**

Gets the comments of this VersionedRemoteProcessGroup. The user-supplied comments for the component

**Returns** The comments of this VersionedRemoteProcessGroup.

**Return type** str

#### **communications\_timeout**

Gets the communications\_timeout of this VersionedRemoteProcessGroup. The time period used for the timeout when communicating with the target.

**Returns** The communications\_timeout of this VersionedRemoteProcessGroup.

**Return type** str



**component\_type**

Gets the component\_type of this VersionedRemoteProcessGroup.

**Returns** The component\_type of this VersionedRemoteProcessGroup.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedRemoteProcessGroup. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedRemoteProcessGroup.

**Return type** str

**identifier**

Gets the identifier of this VersionedRemoteProcessGroup. The component's unique identifier

**Returns** The identifier of this VersionedRemoteProcessGroup.

**Return type** str

**input\_ports**

Gets the input\_ports of this VersionedRemoteProcessGroup. A Set of Input Ports that can be connected to, in order to send data to the remote NiFi instance

**Returns** The input\_ports of this VersionedRemoteProcessGroup.

**Return type** list[*VersionedRemoteGroupPort*]

**instance\_identifier**

Gets the instance\_identifier of this VersionedRemoteProcessGroup. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedRemoteProcessGroup.

**Return type** str

**local\_network\_interface**

Gets the local\_network\_interface of this VersionedRemoteProcessGroup. The local network interface to send/receive data. If not specified, any local address is used. If clustered, all nodes must have an interface with this identifier.

**Returns** The local\_network\_interface of this VersionedRemoteProcessGroup.

**Return type** str

**name**

Gets the name of this VersionedRemoteProcessGroup. The component's name

**Returns** The name of this VersionedRemoteProcessGroup.

**Return type** str

**output\_ports**

Gets the output\_ports of this VersionedRemoteProcessGroup. A Set of Output Ports that can be connected to, in order to pull data from the remote NiFi instance

**Returns** The output\_ports of this VersionedRemoteProcessGroup.

**Return type** list[*VersionedRemoteGroupPort*]

**position**

Gets the position of this VersionedRemoteProcessGroup. The component's position on the graph

**Returns** The position of this VersionedRemoteProcessGroup.

**Return type** Position

**proxy\_host**

Gets the proxy\_host of this VersionedRemoteProcessGroup.

**Returns** The proxy\_host of this VersionedRemoteProcessGroup.

**Return type** str

**proxy\_port**

Gets the proxy\_port of this VersionedRemoteProcessGroup.

**Returns** The proxy\_port of this VersionedRemoteProcessGroup.

**Return type** int

**proxy\_user**

Gets the proxy\_user of this VersionedRemoteProcessGroup.

**Returns** The proxy\_user of this VersionedRemoteProcessGroup.

**Return type** str

**swagger\_types** = {'comments': 'str', 'communications\_timeout': 'str', 'component\_type'

**target\_uri**

Gets the target\_uri of this VersionedRemoteProcessGroup. [DEPRECATED] The target URI of the remote process group. If target uri is not set, but uris are set, then returns the first uri in the uris. If neither target uri nor uris are set, then returns null.

**Returns** The target\_uri of this VersionedRemoteProcessGroup.

**Return type** str

**target\_uris**

Gets the target\_uris of this VersionedRemoteProcessGroup. The target URIs of the remote process group. If target uris is not set but target uri is set, then returns the single target uri. If neither target uris nor target uri is set, then returns null.

**Returns** The target\_uris of this VersionedRemoteProcessGroup.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transport\_protocol**

Gets the transport\_protocol of this VersionedRemoteProcessGroup. The Transport Protocol that is used for Site-to-Site communications

**Returns** The transport\_protocol of this VersionedRemoteProcessGroup.

**Return type** str

**yield\_duration**

Gets the yield\_duration of this VersionedRemoteProcessGroup. When yielding, this amount of time must elapse before the remote process group is scheduled again.

**Returns** The yield\_duration of this VersionedRemoteProcessGroup.

**Return type** str

### 3.3.1.2 Submodules

#### 3.3.1.3 nipyapi.nifi.api\_client module

##### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.nifi.api_client.ApiClient (host=None, header_name=None,  
                                         header_value=None, cookie=None)
```

Bases: object

Generic API client for Swagger client library builds.

Swagger generic API client. This client handles the client- server communication, and is invariant across implementations. Specifics of the methods and models for each application are generated from the Swagger templates.

NOTE: This class is auto generated by the swagger code generator program. Ref: <https://github.com/swagger-api/swagger-codegen> Do not edit the class manually.

##### Parameters

- **host** – The base path for the server to call.
- **header\_name** – a header to pass when making calls to the API.
- **header\_value** – a header value to pass when making calls to the API.

Constructor of the class.

```
NATIVE_TYPES_MAPPING = {'bool': <class 'bool'>, 'date': <class 'datetime.date'>, 'da
```

```
PRIMITIVE_TYPES = (<class 'float'>, <class 'bool'>, <class 'bytes'>, <class 'str'>, <c
```

```
call_api (resource_path, method, path_params=None, query_params=None, header_params=None,  
          body=None, post_params=None, files=None, response_type=None, auth_settings=None,  
          callback=None, _return_http_data_only=None, collection_formats=None,  
          _preload_content=True, _request_timeout=None)
```

Makes the HTTP request (synchronous) and return the deserialized data. To make an async request, define a function for callback.

##### Parameters

- **resource\_path** – Path to method endpoint.
- **method** – Method to call.
- **path\_params** – Path parameters in the url.
- **query\_params** – Query parameters in the url.
- **header\_params** – Header parameters to be placed in the request header.
- **body** – Request body.
- **dict** (*files*) – Request post form parameters, for *application/x-www-form-urlencoded*, *multipart/form-data*.
- **list** (*auth\_settings*) – Auth Settings names for the request.
- **response** – Response data type.

- **dict** – key -> filename, value -> filepath, for *multipart/form-data*.
- **function** (*callback*) – Callback function for asynchronous request. If provide this parameter, the request will be called asynchronously.
- **\_return\_http\_data\_only** – response data without head status code and headers
- **collection\_formats** – dict of collection formats for path, query, header, and post parameters.
- **\_preload\_content** – if False, the urllib3.HTTPResponse object will be returned without reading/decoding response data. Default is True.
- **\_request\_timeout** – timeout setting for this request. If one number provided, it will be total request timeout. It can also be a pair (tuple) of (connection, read) timeouts.

**Returns** If provide parameter callback, the request will be called asynchronously. The method will return the request thread. If parameter callback is None, then the method will return the response directly.

**deserialize** (*response, response\_type*)

Deserializes response into an object.

**Parameters**

- **response** – RESTResponse object to be deserialized.
- **response\_type** – class literal for deserialized object, or string of class name.

**Returns** deserialized object.

**deserialize\_model** (*data, klass*)

Deserializes list or dict to model.

**Parameters**

- **data** – dict, list.
- **klass** – class literal.

**Returns** model object.

**parameters\_to\_tuples** (*params, collection\_formats*)

Get parameters as list of tuples, formatting collections.

**Parameters**

- **params** – Parameters as dict or list of two-tuples
- **collection\_formats** (*dict*) – Parameter collection formats

**Returns** Parameters as list of tuples, collections formatted

**prepare\_post\_parameters** (*post\_params=None, files=None*)

Builds form parameters.

**Parameters**

- **post\_params** – Normal form parameters.
- **files** – File parameters.

**Returns** Form parameters with files.

**request** (*method, url, query\_params=None, headers=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

Makes the HTTP request using RESTClient.

**sanitize\_for\_serialization** (*obj*)

Builds a JSON POST object.

If *obj* is None, return None. If *obj* is str, int, long, float, bool, return directly. If *obj* is datetime.datetime, datetime.date

convert to string in iso8601 format.

If *obj* is list, sanitize each element in the list. If *obj* is dict, return the dict. If *obj* is swagger model, return the properties dict.

**Parameters** *obj* – The data to serialize.

**Returns** The serialized form of data.

**select\_header\_accept** (*accepts*)

Returns *Accept* based on an array of accepts provided.

**Parameters** *accepts* – List of headers.

**Returns** Accept (e.g. application/json).

**select\_header\_content\_type** (*content\_types*)

Returns *Content-Type* based on an array of content\_types provided.

**Parameters** *content\_types* – List of content-types.

**Returns** Content-Type (e.g. application/json).

**set\_default\_header** (*header\_name, header\_value*)**update\_params\_for\_auth** (*headers, querys, auth\_settings*)

Updates header and query params based on authentication setting.

**Parameters**

- **headers** – Header parameters dict to be updated.
- **querys** – Query parameters tuple list to be updated.
- **auth\_settings** – Authentication setting identifiers list.

**user\_agent**

Gets user agent.

**3.3.1.4 nipyapi.nifi.configuration module****NiFi Rest API**

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

`nipyapi.nifi.configuration.Configuration()`

`nipyapi.nifi.configuration.singleton(cls, *args, **kw)`

### 3.3.1.5 nipyapi.nifi.rest module

#### NiFi Rest API

The Rest API provides programmatic access to command and control a NiFi instance in real time. Start and stop processors, monitor queues, query provenance data, and more. Each endpoint below includes a description, definitions of the expected input and output, potential response codes, and the authorizations required to invoke each service.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**exception** `nipyapi.nifi.rest.ApiException` (*status=None, reason=None, http\_resp=None*)

Bases: `Exception`

**class** `nipyapi.nifi.rest.RESTClientObject` (*pools\_size=4, maxsize=4*)

Bases: `object`

**DELETE** (*url, headers=None, query\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

**GET** (*url, headers=None, query\_params=None, \_preload\_content=True, \_request\_timeout=None*)

**HEAD** (*url, headers=None, query\_params=None, \_preload\_content=True, \_request\_timeout=None*)

**OPTIONS** (*url, headers=None, query\_params=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

**PATCH** (*url, headers=None, query\_params=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

**POST** (*url, headers=None, query\_params=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

**PUT** (*url, headers=None, query\_params=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

**request** (*method, url, query\_params=None, headers=None, body=None, post\_params=None, \_preload\_content=True, \_request\_timeout=None*)

#### Parameters

- **method** – http request method
- **url** – http request url
- **query\_params** – query parameters in the url
- **headers** – http request headers
- **body** – request json body, for *application/json*
- **post\_params** – request post parameters, *application/x-www-form-urlencoded* and *multipart/form-data*
- **\_preload\_content** – if False, the `urllib3.HTTPResponse` object will be returned without reading/decoding response data. Default is True.
- **\_request\_timeout** – timeout setting for this request. If one number provided, it will be total request timeout. It can also be a pair (tuple) of (connection, read) timeouts.

**class** `nipyapi.nifi.rest.RESTResponse` (*resp*)

Bases: `io.IOBase`

**getheader** (*name, default=None*)

Returns a given response header.

**getheaders()**

Returns a dictionary of the response headers.

## 3.3.2 NiFi-Registry Swagger Client

### 3.3.2.1 Subpackages

**nipyapi.registry.apis package**

**Submodules**

**nipyapi.registry.apis.access\_api module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.access\_api.**AccessApi** (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_access\_token\_by\_trying\_all\_providers** (*\*\*kwargs*)

Create token trying all providers Creates a token for accessing the REST API via auto-detected method of verifying client identity claim credentials. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_by\_trying\_all\_providers(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_by\_trying\_all\_providers\_with\_http\_info** (*\*\*kwargs*)

Create token trying all providers Creates a token for accessing the REST API via auto-detected method of verifying client identity claim credentials. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_by\_trying\_all\_providers\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_basic\_auth\_credentials** (\*\*kwargs)

Create token using basic auth Creates a token for accessing the REST API via username/password. The user credentials must be passed in standard HTTP Basic Auth format. That is: 'Authorization: Basic <credentials>', where <credentials> is the base64 encoded value of '<username>:<password>'. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_basic\_auth\_credentials(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_basic\_auth\_credentials\_with\_http\_info** (\*\*kwargs)

Create token using basic auth Creates a token for accessing the REST API via username/password. The user credentials must be passed in standard HTTP Basic Auth format. That is: 'Authorization: Basic <credentials>', where <credentials> is the base64 encoded value of '<username>:<password>'. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_basic\_auth\_credentials\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_identity\_provider\_credentials** (\*\*kwargs)

Create token using identity provider Creates a token for accessing the REST API via a custom identity provider. The user credentials must be passed in a format understood by the custom identity provider, e.g., a third-party auth token in an HTTP header. The exact format of the user credentials expected by the custom identity provider can be discovered by 'GET /access/token/identity-provider/usage'. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_identity\_provider\_credentials(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_identity\_provider\_credentials\_with\_http\_info** (\*\*kwargs)

Create token using identity provider Creates a token for accessing the REST API via a custom identity provider. The user credentials must be passed in a format understood by the custom identity provider, e.g., a third-party auth token in an HTTP header. The exact format of the user credentials expected by the custom identity provider can be discovered by 'GET /access/token/identity-provider/usage'. The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and



comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_identity\_provider\_credentials\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_kerberos\_ticket** (*\*\*kwargs*)

Create token using kerberos Creates a token for accessing the REST API via Kerberos Service Tickets or SPNEGO Tokens (which includes Kerberos Service Tickets). The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_kerberos\_ticket(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**create\_access\_token\_using\_kerberos\_ticket\_with\_http\_info** (*\*\*kwargs*)

Create token using kerberos Creates a token for accessing the REST API via Kerberos Service Tickets or SPNEGO Tokens (which includes Kerberos Service Tickets). The token returned is formatted as a JSON Web Token (JWT). The token is base64 encoded and comprised of three parts. The header, the body, and the signature. The expiration of the token is a contained within the body. The token can be used in the Authorization header in the format 'Authorization: Bearer <token>'. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_token\_using\_kerberos\_ticket\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**get\_access\_status** (*\*\*kwargs*)

Get access status Returns the current client's authenticated identity and permissions to top-level resources This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_status(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** CurrentUser If the method is called asynchronously, returns the request thread.

**get\_access\_status\_with\_http\_info** (*\*\*kwargs*)

Get access status Returns the current client's authenticated identity and permissions to top-level resources This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving

```
the response. >>> def callback_function(response): >>> pprint(response) >>> >>> thread =
api.get_access_status_with_http_info(callback=callback_function)
```

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `CurrentUser` If the method is called asynchronously, returns the request thread.

**get\_identity\_provider\_usage\_instructions** (*\*\*kwargs*)

Get identity provider usage Provides a description of how the currently configured identity provider expects credentials to be passed to POST /access/token/identity-provider This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_identity\_provider\_usage\_instructions(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `str` If the method is called asynchronously, returns the request thread.

**get\_identity\_provider\_usage\_instructions\_with\_http\_info** (*\*\*kwargs*)

Get identity provider usage Provides a description of how the currently configured identity provider expects credentials to be passed to POST /access/token/identity-provider This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_identity\_provider\_usage\_instructions\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `str` If the method is called asynchronously, returns the request thread.

**log\_out** (*\*\*kwargs*)

**Performs a logout for other providers that have been issued a JWT.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `None` If the method is called asynchronously, returns the request thread.

**log\_out\_with\_http\_info** (*\*\*kwargs*)

**Performs a logout for other providers that have been issued a JWT.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.log\_out\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `None` If the method is called asynchronously, returns the request thread.

**oidc\_callback** (*\*\*kwargs*)

**Redirect/callback URI for processing the result of the OpenId Connect login sequence.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.oidc_callback(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**oidc\_callback\_with\_http\_info** (\*\*kwargs)

**Redirect/callback URI for processing the result of the OpenId Connect login sequence.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.oidc_callback_with_http_info(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**oidc\_exchange** (\*\*kwargs)

**Retrieves a JWT following a successful login sequence using the configured OpenId Connect provider.**

NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.oidc_exchange(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**oidc\_exchange\_with\_http\_info** (\*\*kwargs)

**Retrieves a JWT following a successful login sequence using the configured OpenId Connect provider.**

NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.oidc_exchange_with_http_info(callback=callback_function)
```

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** str If the method is called asynchronously, returns the request thread.

**oidc\_logout** (\*\*kwargs)

**Performs a logout in the OpenId Provider.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.oidc\_logout(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**oidc\_logout\_with\_http\_info** (\*\*kwargs)

**Performs a logout in the OpenId Provider.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.oidc\_logout\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**oidc\_request** (\*\*kwargs)

**Initiates a request to authenticate through the configured OpenId Connect provider.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.oidc\_request(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**oidc\_request\_with\_http\_info** (\*\*kwargs)

**Initiates a request to authenticate through the configured OpenId Connect provider.** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.oidc\_request\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** None If the method is called asynchronously, returns the request thread.

**test\_identity\_provider\_recognizes\_credentials\_format** (\*\*kwargs)

Test identity provider Tests the format of the credentials against this identity provider without performing authentication on the credentials to validate them. The user credentials should be passed in a format understood by the custom identity provider as defined by ‘GET /access/token/identity-provider/usage’. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.test\_identity\_provider\_recognizes\_credentials\_format(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** *str* If the method is called asynchronously, returns the request thread.

**test\_identity\_provider\_recognizes\_credentials\_format\_with\_http\_info** (*\*\*kwargs*)

Test identity provider Tests the format of the credentials against this identity provider without performing authentication on the credentials to validate them. The user credentials should be passed in a format understood by the custom identity provider as defined by ‘GET /access/token/identity-provider/usage’. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.test\_identity\_provider\_recognizes\_credentials\_format\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** *str* If the method is called asynchronously, returns the request thread.

### nipyapi.registry.apis.bucket\_flows\_api module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.bucket\_flows\_api.**BucketFlowsApi** (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually. Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_flow** (*bucket\_id*, *body*, *\*\*kwargs*)

Create flow Creates a flow in the given bucket. The flow id is created by the server and populated in the returned entity. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow(bucket\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **body** (*VersionedFlow*) – The details of the flow to create. (required)

**Returns** *VersionedFlow* If the method is called asynchronously, returns the request thread.

**create\_flow\_version** (*bucket\_id*, *flow\_id*, *body*, *\*\*kwargs*)

Create flow version Creates the next version of a flow. The version number of the object being created must be the next available version integer. Flow versions are immutable after they are created. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow\_version(bucket\_id, flow\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlowSnapshot*) – The new versioned flow snapshot. (required)

**Returns** *VersionedFlowSnapshot* If the method is called asynchronously, returns the request thread.

**create\_flow\_version\_with\_http\_info** (*bucket\_id, flow\_id, body, \*\*kwargs*)

Create flow version Creates the next version of a flow. The version number of the object being created must be the next available version integer. Flow versions are immutable after they are created. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow\_version\_with\_http\_info(bucket\_id, flow\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlowSnapshot*) – The new versioned flow snapshot. (required)

**Returns** *VersionedFlowSnapshot* If the method is called asynchronously, returns the request thread.

**create\_flow\_with\_http\_info** (*bucket\_id, body, \*\*kwargs*)

Create flow Creates a flow in the given bucket. The flow id is created by the server and populated in the returned entity. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_flow\_with\_http\_info(bucket\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **body** (*VersionedFlow*) – The details of the flow to create. (required)

**Returns** *VersionedFlow* If the method is called asynchronously, returns the request thread.

**delete\_flow** (*version, bucket\_id, flow\_id, \*\*kwargs*)

Delete bucket flow Deletes a flow, including all saved versions of that flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_flow(version, bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **bucket\_id** (*str*) – The bucket identifier (required)

- **flow\_id** (*str*) – The flow identifier (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**delete\_flow\_with\_http\_info** (*version*, *bucket\_id*, *flow\_id*, *\*\*kwargs*)

Delete bucket flow Deletes a flow, including all saved versions of that flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_flow\_with\_http\_info(version, bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**export\_versioned\_flow** (*bucket\_id*, *flow\_id*, *version\_number*, *\*\*kwargs*)

Exports specified bucket flow version content Exports the specified version of a flow, including the metadata and content of the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_versioned\_flow(bucket\_id, flow\_id, version\_number, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**export\_versioned\_flow\_with\_http\_info** (*bucket\_id*, *flow\_id*, *version\_number*, *\*\*kwargs*)

Exports specified bucket flow version content Exports the specified version of a flow, including the metadata and content of the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.export\_versioned\_flow\_with\_http\_info(bucket\_id, flow\_id, version\_number, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**get\_flow** (*bucket\_id*, *flow\_id*, *\*\*kwargs*)

Get bucket flow Retrieves the flow with the given id in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**get\_flow\_diff** (*bucket\_id*, *flow\_id*, *version\_a*, *version\_b*, *\*\*kwargs*)

Get bucket flow diff Computes the differences between two given versions of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_diff(bucket\_id, flow\_id, version\_a, version\_b, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_a** (*int*) – The first version number (required)
- **version\_b** (*int*) – The second version number (required)

**Returns** VersionedFlowDifference If the method is called asynchronously, returns the request thread.

**get\_flow\_diff\_with\_http\_info** (*bucket\_id*, *flow\_id*, *version\_a*, *version\_b*, *\*\*kwargs*)

Get bucket flow diff Computes the differences between two given versions of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_diff\_with\_http\_info(bucket\_id, flow\_id, version\_a, version\_b, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_a** (*int*) – The first version number (required)
- **version\_b** (*int*) – The second version number (required)

**Returns** VersionedFlowDifference If the method is called asynchronously, returns the request thread.



**get\_flow\_version** (*bucket\_id, flow\_id, version\_number, \*\*kwargs*)

Get bucket flow version Gets the given version of a flow, including the metadata and content for the version. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_version(bucket\_id, flow\_id, version\_number, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**get\_flow\_version\_with\_http\_info** (*bucket\_id, flow\_id, version\_number, \*\*kwargs*)

Get bucket flow version Gets the given version of a flow, including the metadata and content for the version. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_version\_with\_http\_info(bucket\_id, flow\_id, version\_number, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**get\_flow\_versions** (*bucket\_id, flow\_id, \*\*kwargs*)

Get bucket flow versions Gets summary information for all versions of a flow. Versions are ordered newest->oldest. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_versions(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** list[VersionedFlowSnapshotMetadata] If the method is called asynchronously, returns the request thread.

**get\_flow\_versions\_with\_http\_info** (*bucket\_id, flow\_id, \*\*kwargs*)

Get bucket flow versions Gets summary information for all versions of a flow. Versions are ordered newest->oldest. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_versions\_with\_http\_info(bucket\_id, flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** list[VersionedFlowSnapshotMetadata] If the method is called asynchronously, returns the request thread.

**get\_flow\_with\_http\_info** (*bucket\_id, flow\_id, \*\*kwargs*)

Get bucket flow Retrieves the flow with the given id in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flow\_with\_http\_info(bucket\_id, flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**get\_flows** (*bucket\_id, \*\*kwargs*)

Get bucket flows Retrieves all flows in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flows(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** list[VersionedFlow] If the method is called asynchronously, returns the request thread.

**get\_flows\_with\_http\_info** (*bucket\_id, \*\*kwargs*)

Get bucket flows Retrieves all flows in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_flows\_with\_http\_info(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** list[VersionedFlow] If the method is called asynchronously, returns the request thread.

**get\_latest\_flow\_version** (*bucket\_id, flow\_id, \*\*kwargs*)

Get latest bucket flow version content Gets the latest version of a flow, including the metadata and content of the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_latest\_flow\_version(bucket\_id, flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**get\_latest\_flow\_version\_metadata** (*bucket\_id, flow\_id, \*\*kwargs*)

Get latest bucket flow version metadata Gets the metadata for the latest version of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_latest\_flow\_version\_metadata(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshotMetadata If the method is called asynchronously, returns the request thread.

**get\_latest\_flow\_version\_metadata\_with\_http\_info** (*bucket\_id, flow\_id, \*\*kwargs*)

Get latest bucket flow version metadata Gets the metadata for the latest version of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_latest\_flow\_version\_metadata\_with\_http\_info(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshotMetadata If the method is called asynchronously, returns the request thread.

**get\_latest\_flow\_version\_with\_http\_info** (*bucket\_id, flow\_id, \*\*kwargs*)

Get latest bucket flow version content Gets the latest version of a flow, including the metadata and content of the flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_latest\_flow\_version\_with\_http\_info(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**import\_versioned\_flow** (*bucket\_id, flow\_id, \*\*kwargs*)

Import flow version Import the next version of a flow. The version number of the object being created will be the next available version integer. Flow versions are immutable after they are created.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_versioned\_flow(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlowSnapshot*) – file
- **comments** (*str*) –

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**import\_versioned\_flow\_with\_http\_info** (*bucket\_id, flow\_id, \*\*kwargs*)

Import flow version Import the next version of a flow. The version number of the object being created will be the next available version integer. Flow versions are immutable after they are created. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.import\_versioned\_flow\_with\_http\_info(bucket\_id, flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlowSnapshot*) – file
- **comments** (*str*) –

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**update\_flow** (*bucket\_id, flow\_id, body, \*\*kwargs*)

Update bucket flow Updates the flow with the given id in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_flow(bucket\_id, flow\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlow*) – The updated flow (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**update\_flow\_with\_http\_info** (*bucket\_id, flow\_id, body, \*\*kwargs*)

Update bucket flow Updates the flow with the given id in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a

*callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_flow\_with\_http\_info(bucket\_id, flow\_id, body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **flow\_id** (*str*) – The flow identifier (required)
- **body** (*VersionedFlow*) – The updated flow (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

### nipyapi.registry.apis.buckets\_api module

#### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.buckets\_api.BucketsApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_bucket** (*body, \*\*kwargs*)

Create bucket

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_bucket(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*Bucket*) – The bucket to create (required)

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**create\_bucket\_with\_http\_info** (*body, \*\*kwargs*)

Create bucket

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_bucket\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*Bucket*) – The bucket to create (required)

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**delete\_bucket** (*version*, *bucket\_id*, *\*\*kwargs*)

Delete bucket Deletes the bucket with the given id, along with all objects stored in the bucket This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_bucket(version, bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**delete\_bucket\_with\_http\_info** (*version*, *bucket\_id*, *\*\*kwargs*)

Delete bucket Deletes the bucket with the given id, along with all objects stored in the bucket This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.delete\_bucket\_with\_http\_info(version, bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**get\_available\_bucket\_fields** (*\*\*kwargs*)

Get bucket fields Retrieves bucket field names for searching or sorting on buckets. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_bucket\_fields(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** Fields If the method is called asynchronously, returns the request thread.

**get\_available\_bucket\_fields\_with\_http\_info** (*\*\*kwargs*)

Get bucket fields Retrieves bucket field names for searching or sorting on buckets. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_bucket\_fields\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** Fields If the method is called asynchronously, returns the request thread.

**get\_bucket** (*bucket\_id*, *\*\*kwargs*)

Get bucket Gets the bucket with the given id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bucket(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**get\_bucket\_with\_http\_info** (*bucket\_id*, *\*\*kwargs*)

Get bucket Gets the bucket with the given id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_bucket\_with\_http\_info(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** Bucket If the method is called asynchronously, returns the request thread.

**get\_buckets** (*\*\*kwargs*)

Get all buckets The returned list will include only buckets for which the user is authorized.If the user is not authorized for any buckets, this returns an empty list. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_buckets(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[Bucket] If the method is called asynchronously, returns the request thread.

**get\_buckets\_with\_http\_info** (*\*\*kwargs*)

Get all buckets The returned list will include only buckets for which the user is authorized.If the user is not authorized for any buckets, this returns an empty list. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_buckets\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[Bucket] If the method is called asynchronously, returns the request thread.

**update\_bucket** (*bucket\_id*, *body*, *\*\*kwargs*)

Update bucket Updates the bucket with the given id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_bucket(bucket\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

- **body** (`Bucket`) – The updated bucket (required)

**Returns** `Bucket` If the method is called asynchronously, returns the request thread.

**update\_bucket\_with\_http\_info** (*bucket\_id, body, \*\*kwargs*)

Update bucket Updates the bucket with the given id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_bucket\_with\_http\_info(bucket\_id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)
- **body** (`Bucket`) – The updated bucket (required)

**Returns** `Bucket` If the method is called asynchronously, returns the request thread.

## nipyapi.registry.apis.flows\_api module

### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.flows\_api.**FlowsApi** (*api\_client=None*)  
Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_available\_flow\_fields** (*\*\*kwargs*)

Get flow fields Retrieves the flow field names that can be used for searching or sorting on flows. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_flow\_fields(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `Fields` If the method is called asynchronously, returns the request thread.

**get\_available\_flow\_fields\_with\_http\_info** (*\*\*kwargs*)

Get flow fields Retrieves the flow field names that can be used for searching or sorting on flows. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_flow\_fields\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** `Fields` If the method is called asynchronously, returns the request thread.



**global\_get\_flow** (*flow\_id*, *\*\*kwargs*)

Get flow Gets a flow by id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**global\_get\_flow\_version** (*flow\_id*, *version\_number*, *\*\*kwargs*)

Get flow version Gets the given version of a flow, including metadata and flow content. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow\_version(flow\_id, version\_number, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**global\_get\_flow\_version\_with\_http\_info** (*flow\_id*, *version\_number*, *\*\*kwargs*)

Get flow version Gets the given version of a flow, including metadata and flow content. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow\_version\_with\_http\_info(flow\_id, version\_number, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)
- **version\_number** (*int*) – The version number (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**global\_get\_flow\_versions** (*flow\_id*, *\*\*kwargs*)

Get flow versions Gets summary information for all versions of a given flow. Versions are ordered newest->oldest. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow\_versions(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** list[VersionedFlowSnapshotMetadata] If the method is called asynchronously, returns the request thread.

**global\_get\_flow\_versions\_with\_http\_info** (*flow\_id*, *\*\*kwargs*)

Get flow versions Gets summary information for all versions of a given flow. Versions are ordered newest->oldest. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow\_versions\_with\_http\_info(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** list[VersionedFlowSnapshotMetadata] If the method is called asynchronously, returns the request thread.

**global\_get\_flow\_with\_http\_info** (*flow\_id*, *\*\*kwargs*)

Get flow Gets a flow by id. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_flow\_with\_http\_info(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlow If the method is called asynchronously, returns the request thread.

**global\_get\_latest\_flow\_version** (*flow\_id*, *\*\*kwargs*)

Get latest flow version Gets the latest version of a flow, including metadata and flow content. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_latest\_flow\_version(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

**global\_get\_latest\_flow\_version\_metadata** (*flow\_id*, *\*\*kwargs*)

Get latest flow version metadata Gets the metadata for the latest version of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_latest\_flow\_version\_metadata(flow\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshotMetadata If the method is called asynchronously, returns the request thread.

**global\_get\_latest\_flow\_version\_metadata\_with\_http\_info** (*flow\_id*, *\*\*kwargs*)

Get latest flow version metadata Gets the metadata for the latest version of a flow. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_latest\_flow\_version\_metadata\_with\_http\_info(flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshotMetadata If the method is called asynchronously, returns the request thread.

**global\_get\_latest\_flow\_version\_with\_http\_info** (*flow\_id*, *\*\*kwargs*)

Get latest flow version Gets the latest version of a flow, including metadata and flow content. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.global\_get\_latest\_flow\_version\_with\_http\_info(flow\_id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **flow\_id** (*str*) – The flow identifier (required)

**Returns** VersionedFlowSnapshot If the method is called asynchronously, returns the request thread.

## nipyapi.registry.apis.items\_api module

### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.items\_api.ItemsApi (*api\_client=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**get\_available\_bucket\_item\_fields** (*\*\*kwargs*)

Get item fields Retrieves the item field names for searching or sorting on bucket items. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_bucket\_item\_fields(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** Fields If the method is called asynchronously, returns the request thread.

**get\_available\_bucket\_item\_fields\_with\_http\_info** (*\*\*kwargs*)

Get item fields Retrieves the item field names for searching or sorting on bucket items.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_available\_bucket\_item\_fields\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** Fields If the method is called asynchronously, returns the request thread.

**get\_items** (*\*\*kwargs*)

Get all items Get items across all buckets. The returned items will include only items from buckets for which the user is authorized. If the user is not authorized to any buckets, an empty list will be returned. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_items(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[BucketItem] If the method is called asynchronously, returns the request thread.

**get\_items\_in\_bucket** (*bucket\_id*, *\*\*kwargs*)

Get bucket items Gets the items located in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_items\_in\_bucket(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** list[BucketItem] If the method is called asynchronously, returns the request thread.

**get\_items\_in\_bucket\_with\_http\_info** (*bucket\_id*, *\*\*kwargs*)

Get bucket items Gets the items located in the given bucket. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_items\_in\_bucket\_with\_http\_info(bucket\_id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **bucket\_id** (*str*) – The bucket identifier (required)

**Returns** list[BucketItem] If the method is called asynchronously, returns the request thread.

**get\_items\_with\_http\_info** (*\*\*kwargs*)

Get all items Get items across all buckets. The returned items will include only items from buckets for which the user is authorized. If the user is not authorized to any buckets, an empty list will be returned. This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_items\_with\_http\_info(callback=callback\_function)

**Parameters** *function* (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[BucketItem] If the method is called asynchronously, returns the request thread.

## nipyapi.registry.apis.policies\_api module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.apis.policies\_api.PoliciesApi (api\_client=None)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_access\_policy** (body, \*\*kwargs)

Create access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_policy(body, callback=callback\_function)

### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*AccessPolicy*) – The access policy configuration details. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

**create\_access\_policy\_with\_http\_info** (body, \*\*kwargs)

Create access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_access\_policy\_with\_http\_info(body, callback=callback\_function)

### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*AccessPolicy*) – The access policy configuration details. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

**get\_access\_policies** (\*\*kwargs)

Get all access policies

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policies(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[AccessPolicy] If the method is called asynchronously, returns the request thread.

**get\_access\_policies\_with\_http\_info** (\*\*kwargs)

Get all access policies

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policies\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[AccessPolicy] If the method is called asynchronously, returns the request thread.

**get\_access\_policy** (*id*, **\*\*kwargs**)

Get access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

**get\_access\_policy\_for\_resource** (*action*, *resource*, **\*\*kwargs**)

Get access policy for resource Gets an access policy for the specified action and resource This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_for\_resource(action, resource, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **action** (*str*) – The request action. (required)
- **resource** (*str*) – The resource of the policy. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

**get\_access\_policy\_for\_resource\_with\_http\_info** (*action*, *resource*, **\*\*kwargs**)

Get access policy for resource Gets an access policy for the specified action and resource This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_for\_resource\_with\_http\_info(action, resource, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **action** (*str*) – The request action. (required)
- **resource** (*str*) – The resource of the policy. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

**get\_access\_policy\_with\_http\_info** (*id*, **\*\*kwargs**)

Get access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_access\_policy\_with\_http\_info(id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

#### get\_resources (\*\*kwargs)

Get available resources Gets the available resources that support access/authorization policies This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_resources(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[Resource] If the method is called asynchronously, returns the request thread.

#### get\_resources\_with\_http\_info (\*\*kwargs)

Get available resources Gets the available resources that support access/authorization policies This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_resources\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[Resource] If the method is called asynchronously, returns the request thread.

#### remove\_access\_policy (version, id, \*\*kwargs)

Delete access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_access\_policy(version, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The access policy id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** AccessPolicy If the method is called asynchronously, returns the request thread.

#### remove\_access\_policy\_with\_http\_info (version, id, \*\*kwargs)

Delete access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_access\_policy\_with\_http\_info(version, id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The access policy id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** `AccessPolicy` If the method is called asynchronously, returns the request thread.

**update\_access\_policy** (*id*, *body*, *\*\*kwargs*)

Update access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_access\_policy(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **body** (`AccessPolicy`) – The access policy configuration details. (required)

**Returns** `AccessPolicy` If the method is called asynchronously, returns the request thread.

**update\_access\_policy\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

Update access policy

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_access\_policy\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The access policy id. (required)
- **body** (`AccessPolicy`) – The access policy configuration details. (required)

**Returns** `AccessPolicy` If the method is called asynchronously, returns the request thread.

## nipyapi.registry.apis.tenants\_api module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.apis.tenants_api.TenantsApi (api_client=None)
    Bases: object
```



NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.  
Ref: <https://github.com/swagger-api/swagger-codegen>

**create\_user** (*body*, *\*\*kwargs*)

**Create user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*User*) – The user configuration details. (required)

**Returns** User If the method is called asynchronously, returns the request thread.

**create\_user\_group** (*body*, *\*\*kwargs*)

**Create user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user\_group(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*UserGroup*) – The user group configuration details. (required)

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**create\_user\_group\_with\_http\_info** (*body*, *\*\*kwargs*)

**Create user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user\_group\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **body** (*UserGroup*) – The user group configuration details. (required)

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**create\_user\_with\_http\_info** (*body*, *\*\*kwargs*)

**Create user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.create\_user\_with\_http\_info(body, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)

- **body** (*User*) – The user configuration details. (required)

**Returns** User If the method is called asynchronously, returns the request thread.

**get\_user** (*id*, *\*\*kwargs*)

**Get user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)

**Returns** User If the method is called asynchronously, returns the request thread.

**get\_user\_group** (*id*, *\*\*kwargs*)

**Get user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_group(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**get\_user\_group\_with\_http\_info** (*id*, *\*\*kwargs*)

**Get user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_group\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**get\_user\_groups** (*\*\*kwargs*)

**Get user groups** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_groups(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[UserGroup] If the method is called asynchronously, returns the request thread.

**get\_user\_groups\_with\_http\_info** (\*\*kwargs)

**Get user groups** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_groups\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[UserGroup] If the method is called asynchronously, returns the request thread.

**get\_user\_with\_http\_info** (id, \*\*kwargs)

**Get user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_user\_with\_http\_info(id, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)

**Returns** User If the method is called asynchronously, returns the request thread.

**get\_users** (\*\*kwargs)

**Get all users** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_users(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[User] If the method is called asynchronously, returns the request thread.

**get\_users\_with\_http\_info** (\*\*kwargs)

**Get all users** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.get\_users\_with\_http\_info(callback=callback\_function)

**Parameters** **function** (*callback*) – The callback function for asynchronous request. (optional)

**Returns** list[User] If the method is called asynchronously, returns the request thread.

**remove\_user** (version, id, \*\*kwargs)

**Delete user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user(version, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The user id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** User If the method is called asynchronously, returns the request thread.

**remove\_user\_group** (*version, id, \*\*kwargs*)

**Delete user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user\_group(version, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The user group id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**remove\_user\_group\_with\_http\_info** (*version, id, \*\*kwargs*)

**Delete user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.remove\_user\_group\_with\_http\_info(version, id, callback=callback\_function)

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The user group id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**remove\_user\_with\_http\_info** (*version, id, \*\*kwargs*)

**Delete user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.remove_user_with_http_info(version, id, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **version** (*str*) – The version is used to verify the client is working with the latest version of the entity. (required)
- **id** (*str*) – The user id. (required)
- **client\_id** (*str*) – If the client id is not specified, new one will be generated. This value (whether specified or generated) is included in the response.

**Returns** User If the method is called asynchronously, returns the request thread.

**update\_user** (*id*, *body*, *\*\*kwargs*)

**Update user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **body** (*User*) – The user configuration details. (required)

**Returns** User If the method is called asynchronously, returns the request thread.

**update\_user\_group** (*id*, *body*, *\*\*kwargs*)

**Update user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user_group(id, body, callback=callback_function)
```

#### Parameters

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **body** (*UserGroup*) – The user group configuration details. (required)

**Returns** UserGroup If the method is called asynchronously, returns the request thread.

**update\_user\_group\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

**Update user group** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. 

```
>>> def callback_function(response): >>> pprint(response) >>> >>> thread = api.update_user_group_with_http_info(id, body, callback=callback_function)
```

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user group id. (required)
- **body** (*UserGroup*) – The user group configuration details. (required)

**Returns** *UserGroup* If the method is called asynchronously, returns the request thread.

**update\_user\_with\_http\_info** (*id*, *body*, *\*\*kwargs*)

**Update user** NOTE: This endpoint is subject to change as NiFi Registry and its REST API evolve.

This method makes a synchronous HTTP request by default. To make an asynchronous HTTP request, please define a *callback* function to be invoked when receiving the response. >>> def callback\_function(response): >>> pprint(response) >>> >>> thread = api.update\_user\_with\_http\_info(id, body, callback=callback\_function)

**Parameters**

- **function** (*callback*) – The callback function for asynchronous request. (optional)
- **id** (*str*) – The user id. (required)
- **body** (*User*) – The user configuration details. (required)

**Returns** *User* If the method is called asynchronously, returns the request thread.

**nipyapi.registry.models package****Submodules****nipyapi.registry.models.access\_policy module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.access_policy.AccessPolicy (identifier=None, re-  
source=None, ac-  
tion=None, config-  
urable=None, revi-  
sion=None, users=None,  
user_groups=None)
```

Bases: *object*

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicy - a model defined in Swagger

**action**

Gets the action of this AccessPolicy. The action associated with this access policy.

**Returns** The action of this AccessPolicy.

**Return type** *str*

```
attribute_map = {'action': 'action', 'configurable': 'configurable', 'identifier':
```

**configurable**

Gets the configurable of this AccessPolicy. Indicates if this access policy is configurable, based on which Authorizer has been configured to manage it.

**Returns** The configurable of this AccessPolicy.

**Return type** bool

**identifier**

Gets the identifier of this AccessPolicy. The id of the policy. Set by server at creation time.

**Returns** The identifier of this AccessPolicy.

**Return type** str

**resource**

Gets the resource of this AccessPolicy. The resource for this access policy.

**Returns** The resource of this AccessPolicy.

**Return type** str

**revision**

Gets the revision of this AccessPolicy. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this AccessPolicy.

**Return type** RevisionInfo

**swagger\_types** = {'action': 'str', 'configurable': 'bool', 'identifier': 'str', 'res

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**user\_groups**

Gets the user\_groups of this AccessPolicy. The set of user group IDs associated with this access policy.

**Returns** The user\_groups of this AccessPolicy.

**Return type** list[Tenant]

**users**

Gets the users of this AccessPolicy. The set of user IDs associated with this access policy.

**Returns** The users of this AccessPolicy.

**Return type** list[Tenant]

**nipyapi.registry.models.access\_policy\_summary module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.access_policy_summary.AccessPolicySummary (identifier=None,
                                                                           re-
                                                                           source=None,
                                                                           ac-
                                                                           tion=None,
                                                                           con-
                                                                           fig-
                                                                           urable=None,
                                                                           re-
                                                                           vi-
                                                                           sion=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

AccessPolicySummary - a model defined in Swagger

#### **action**

Gets the action of this AccessPolicySummary. The action associated with this access policy.

**Returns** The action of this AccessPolicySummary.

**Return type** str

```
attribute_map = {'action': 'action', 'configurable': 'configurable', 'identifier':
```

#### **configurable**

Gets the configurable of this AccessPolicySummary. Indicates if this access policy is configurable, based on which Authorizer has been configured to manage it.

**Returns** The configurable of this AccessPolicySummary.

**Return type** bool

#### **identifier**

Gets the identifier of this AccessPolicySummary. The id of the policy. Set by server at creation time.

**Returns** The identifier of this AccessPolicySummary.

**Return type** str

#### **resource**

Gets the resource of this AccessPolicySummary. The resource for this access policy.

**Returns** The resource of this AccessPolicySummary.

**Return type** str

#### **revision**

Gets the revision of this AccessPolicySummary. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this AccessPolicySummary.

**Return type** RevisionInfo

```
swagger_types = {'action': 'str', 'configurable': 'bool', 'identifier': 'str', 'res
```

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model



## nipyapi.registry.models.batch\_size module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.models.batch\_size.**BatchSize** (*count=None, size=None, duration=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BatchSize - a model defined in Swagger

**attribute\_map** = {'count': 'count', 'duration': 'duration', 'size': 'size'}

**count**

Gets the count of this BatchSize. Preferred number of flow files to include in a transaction.

**Returns** The count of this BatchSize.

**Return type** int

**duration**

Gets the duration of this BatchSize. Preferred amount of time that a transaction should span.

**Returns** The duration of this BatchSize.

**Return type** str

**size**

Gets the size of this BatchSize. Preferred number of bytes to include in a transaction.

**Returns** The size of this BatchSize.

**Return type** str

**swagger\_types** = {'count': 'int', 'duration': 'str', 'size': 'str'}

**to\_dict** ()

Returns the model properties as a dict

**to\_str** ()

Returns the string representation of the model

## nipyapi.registry.models.bucket module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.bucket.Bucket (link=None, identifier=None, name=None,  
created_timestamp=None, description=None, allow_bundle_redeploy=None,  
allow_public_read=None, permissions=None, revision=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Bucket - a model defined in Swagger

**allow\_bundle\_redeploy**

Gets the allow\_bundle\_redeploy of this Bucket. Indicates if this bucket allows the same version of an extension bundle to be redeployed and thus overwrite the existing artifact. By default this is false.

**Returns** The allow\_bundle\_redeploy of this Bucket.

**Return type** bool

**allow\_public\_read**

Gets the allow\_public\_read of this Bucket. Indicates if this bucket allows read access to unauthenticated anonymous users

**Returns** The allow\_public\_read of this Bucket.

**Return type** bool

```
attribute_map = {'allow_bundle_redeploy': 'allowBundleRedeploy', 'allow_public_read':
```

**created\_timestamp**

Gets the created\_timestamp of this Bucket. The timestamp of when the bucket was first created. This is set by the server at creation time.

**Returns** The created\_timestamp of this Bucket.

**Return type** int

**description**

Gets the description of this Bucket. A description of the bucket.

**Returns** The description of this Bucket.

**Return type** str

**identifier**

Gets the identifier of this Bucket. An ID to uniquely identify this object.

**Returns** The identifier of this Bucket.

**Return type** str

**link**

Gets the link of this Bucket. An WebLink to this entity.

**Returns** The link of this Bucket.

**Return type** JaxbLink

**name**

Gets the name of this Bucket. The name of the bucket.

**Returns** The name of this Bucket.

**Return type** str

**permissions**

Gets the permissions of this Bucket. The access that the current user has to this bucket.

**Returns** The permissions of this Bucket.

**Return type** *Permissions*

#### **revision**

Gets the revision of this Bucket. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this Bucket.

**Return type** RevisionInfo

**swagger\_types** = {'allow\_bundle\_redeploy': 'bool', 'allow\_public\_read': 'bool', 'crea

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

### **nipyapi.registry.models.bucket\_item module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.bucket_item.BucketItem(link=None, identifier=None, name=None,
description=None,
bucket_identifier=None,
bucket_name=None, created_timestamp=None,
modified_timestamp=None,
type=None, permissions=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

BucketItem - a model defined in Swagger

**attribute\_map** = {'bucket\_identifier': 'bucketIdentifier', 'bucket\_name': 'bucketName

#### **bucket\_identifier**

Gets the bucket\_identifier of this BucketItem. The identifier of the bucket this items belongs to. This cannot be changed after the item is created.

**Returns** The bucket\_identifier of this BucketItem.

**Return type** str

#### **bucket\_name**

Gets the bucket\_name of this BucketItem. The name of the bucket this items belongs to.

**Returns** The bucket\_name of this BucketItem.

**Return type** str

**created\_timestamp**

Gets the created\_timestamp of this BucketItem. The timestamp of when the item was created, as milliseconds since epoch.

**Returns** The created\_timestamp of this BucketItem.

**Return type** int

**description**

Gets the description of this BucketItem. A description of the item.

**Returns** The description of this BucketItem.

**Return type** str

**identifier**

Gets the identifier of this BucketItem. An ID to uniquely identify this object.

**Returns** The identifier of this BucketItem.

**Return type** str

**link**

Gets the link of this BucketItem. An WebLink to this entity.

**Returns** The link of this BucketItem.

**Return type** JaxbLink

**modified\_timestamp**

Gets the modified\_timestamp of this BucketItem. The timestamp of when the item was last modified, as milliseconds since epoch.

**Returns** The modified\_timestamp of this BucketItem.

**Return type** int

**name**

Gets the name of this BucketItem. The name of the item.

**Returns** The name of this BucketItem.

**Return type** str

**permissions**

Gets the permissions of this BucketItem. The access that the current user has to the bucket containing this item.

**Returns** The permissions of this BucketItem.

**Return type** *Permissions*

**swagger\_types** = {'bucket\_identifier': 'str', 'bucket\_name': 'str', 'created\_timestamp': 'int', 'description': 'str', 'link': 'JaxbLink', 'modified\_timestamp': 'int', 'name': 'str', 'permissions': 'Permissions', 'type': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this BucketItem. The type of item.

**Returns** The type of this BucketItem.

**Return type** str

## nipyapi.registry.models.bundle module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```

class nipyapi.registry.models.bundle.Bundle (group=None,      artifact=None,      ver-
                                             sion=None)

    Bases: object

    NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

    Bundle - a model defined in Swagger

    artifact
        Gets the artifact of this Bundle. The artifact of the bundle

        Returns The artifact of this Bundle.

        Return type str

    attribute_map = {'artifact':  'artifact', 'group':  'group', 'version':  'version'}

    group
        Gets the group of this Bundle. The group of the bundle

        Returns The group of this Bundle.

        Return type str

    swagger_types = {'artifact':  'str', 'group':  'str', 'version':  'str'}

    to_dict ()
        Returns the model properties as a dict

    to_str ()
        Returns the string representation of the model

    version
        Gets the version of this Bundle. The version of the bundle

        Returns The version of this Bundle.

        Return type str

```

## nipyapi.registry.models.connectable\_component module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.connectable_component.ConnectableComponent (id=None,  
type=None,  
group_id=None,  
name=None,  
com-  
ments=None,  
in-  
stance_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ConnectableComponent - a model defined in Swagger

```
attribute_map = {'comments':  'comments', 'group_id':  'groupId', 'id':  'id', 'instance_id':  'instanceId'}
```

**comments**

Gets the comments of this ConnectableComponent. The comments for the connectable component.

**Returns** The comments of this ConnectableComponent.

**Return type** str

**group\_id**

Gets the group\_id of this ConnectableComponent. The id of the group that the connectable component resides in

**Returns** The group\_id of this ConnectableComponent.

**Return type** str

**id**

Gets the id of this ConnectableComponent. The id of the connectable component.

**Returns** The id of this ConnectableComponent.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this ConnectableComponent. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this ConnectableComponent.

**Return type** str

**name**

Gets the name of this ConnectableComponent. The name of the connectable component

**Returns** The name of this ConnectableComponent.

**Return type** str

```
swagger_types = {'comments':  'str', 'group_id':  'str', 'id':  'str', 'instance_id':  'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ConnectableComponent. The type of component the connectable is.

**Returns** The type of this ConnectableComponent.

**Return type** str

### nipyapi.registry.models.controller\_service\_api module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.controller_service_api.ControllerServiceAPI (type=None,
                                                                    bun-
                                                                    dle=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ControllerServiceAPI - a model defined in Swagger

```
attribute_map = {'bundle':  'bundle', 'type':  'type'}
```

**bundle**

Gets the bundle of this ControllerServiceAPI. The details of the artifact that bundled this service interface.

**Returns** The bundle of this ControllerServiceAPI.

**Return type** *Bundle*

```
swagger_types = {'bundle':  'Bundle', 'type':  'str'}
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this ControllerServiceAPI. The fully qualified name of the service interface.

**Returns** The type of this ControllerServiceAPI.

**Return type** str

### nipyapi.registry.models.current\_user module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.current_user.CurrentUser (identity=None,      anony-
                                                                    mous=None,      lo-
                                                                    gin_supported=None,  re-
                                                                    source_permissions=None,
                                                                    oidclo-
                                                                    gin_supported=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

CurrentUser - a model defined in Swagger

**anonymous**

Gets the anonymous of this CurrentUser. Indicates if the current user is anonymous

**Returns** The anonymous of this CurrentUser.

**Return type** bool

**attribute\_map** = {'anonymous': 'anonymous', 'identity': 'identity', 'login\_supported'

**identity**

Gets the identity of this CurrentUser. The identity of the current user

**Returns** The identity of this CurrentUser.

**Return type** str

**login\_supported**

Gets the login\_supported of this CurrentUser. Indicates if the NiFi Registry instance supports logging in

**Returns** The login\_supported of this CurrentUser.

**Return type** bool

**oidclogin\_supported**

Gets the oidclogin\_supported of this CurrentUser. Indicates if the NiFi Registry instance supports logging in with an OIDC provider

**Returns** The oidclogin\_supported of this CurrentUser.

**Return type** bool

**resource\_permissions**

Gets the resource\_permissions of this CurrentUser. The access that the current user has to top level resources

**Returns** The resource\_permissions of this CurrentUser.

**Return type** *ResourcePermissions*

**swagger\_types** = {'anonymous': 'bool', 'identity': 'str', 'login\_supported': 'bool',

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.registry.models.fields module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.models.fields.**Fields** (*fields=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.



Fields - a model defined in Swagger

```
attribute_map = {'fields': 'fields'}
fields
    Gets the fields of this Fields.
    Returns The fields of this Fields.
    Return type list[str]
swagger_types = {'fields': 'list[str]'}
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
```

### nipyapi.registry.models.permissions module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.permissions.Permissions(can_read=None,
                                                    can_write=None,
                                                    can_delete=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Permissions - a model defined in Swagger

```
attribute_map = {'can_delete': 'canDelete', 'can_read': 'canRead', 'can_write': 'canWrite'}
can_delete
    Gets the can_delete of this Permissions. Indicates whether the user can delete a given resource.
    Returns The can_delete of this Permissions.
    Return type bool
can_read
    Gets the can_read of this Permissions. Indicates whether the user can read a given resource.
    Returns The can_read of this Permissions.
    Return type bool
can_write
    Gets the can_write of this Permissions. Indicates whether the user can write a given resource.
    Returns The can_write of this Permissions.
    Return type bool
swagger_types = {'can_delete': 'bool', 'can_read': 'bool', 'can_write': 'bool'}
to_dict()
    Returns the model properties as a dict
```

**to\_str()**  
Returns the string representation of the model

## nipyapi.registry.models.resource module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.models.resource.Resource(*identifier=None, name=None*)

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Resource - a model defined in Swagger

**attribute\_map** = {'identifier': 'identifier', 'name': 'name'}

**identifier**

Gets the identifier of this Resource. The identifier of the resource.

**Returns** The identifier of this Resource.

**Return type** str

**name**

Gets the name of this Resource. The name of the resource.

**Returns** The name of this Resource.

**Return type** str

**swagger\_types** = {'identifier': 'str', 'name': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

## nipyapi.registry.models.resource\_permissions module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** nipyapi.registry.models.resource\_permissions.ResourcePermissions(*buckets=None, tenants=None, policies=None, proxy=None, any\_top\_level\_resource=None*)

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

ResourcePermissions - a model defined in Swagger

#### **any\_top\_level\_resource**

Gets the any\_top\_level\_resource of this ResourcePermissions. The access that the current user has to any top level resources (a logical 'OR' of all other values)

**Returns** The any\_top\_level\_resource of this ResourcePermissions.

**Return type** *Permissions*

**attribute\_map** = {'any\_top\_level\_resource': 'anyTopLevelResource', 'buckets': 'bucket

#### **buckets**

Gets the buckets of this ResourcePermissions. The access that the current user has to the top level /buckets resource of this NiFi Registry (i.e., access to all buckets)

**Returns** The buckets of this ResourcePermissions.

**Return type** *Permissions*

#### **policies**

Gets the policies of this ResourcePermissions. The access that the current user has to the top level /policies resource of this NiFi Registry

**Returns** The policies of this ResourcePermissions.

**Return type** *Permissions*

#### **proxy**

Gets the proxy of this ResourcePermissions. The access that the current user has to the top level /proxy resource of this NiFi Registry

**Returns** The proxy of this ResourcePermissions.

**Return type** *Permissions*

**swagger\_types** = {'any\_top\_level\_resource': 'Permissions', 'buckets': 'Permissions',

#### **tenants**

Gets the tenants of this ResourcePermissions. The access that the current user has to the top level /tenants resource of this NiFi Registry

**Returns** The tenants of this ResourcePermissions.

**Return type** *Permissions*

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

### **nipyapi.registry.models.tenant module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.tenant.Tenant (identifier=None, identity=None, configurable=None, resource_permissions=None, access_policies=None, revision=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

Tenant - a model defined in Swagger

#### **access\_policies**

Gets the access\_policies of this Tenant. The access policies granted to this tenant.

**Returns** The access\_policies of this Tenant.

**Return type** list[*AccessPolicySummary*]

```
attribute_map = {'access_policies': 'accessPolicies', 'configurable': 'configurable'}
```

#### **configurable**

Gets the configurable of this Tenant. Indicates if this tenant is configurable, based on which UserGroup-Provider has been configured to manage it.

**Returns** The configurable of this Tenant.

**Return type** bool

#### **identifier**

Gets the identifier of this Tenant. The computer-generated identifier of the tenant.

**Returns** The identifier of this Tenant.

**Return type** str

#### **identity**

Gets the identity of this Tenant. The human-facing identity of the tenant. This can only be changed if the tenant is configurable.

**Returns** The identity of this Tenant.

**Return type** str

#### **resource\_permissions**

Gets the resource\_permissions of this Tenant. A summary top-level resource access policies granted to this tenant.

**Returns** The resource\_permissions of this Tenant.

**Return type** *ResourcePermissions*

#### **revision**

Gets the revision of this Tenant. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this Tenant.

**Return type** RevisionInfo

```
swagger_types = {'access_policies': 'list[AccessPolicySummary]', 'configurable': 'boolean'}
```

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

## nipyapi.registry.models.user module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.user.User (identifier=None, identity=None, configurable=None, resource_permissions=None, access_policies=None, revision=None, user_groups=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

User - a model defined in Swagger

### **access\_policies**

Gets the access\_policies of this User. The access policies granted to this tenant.

**Returns** The access\_policies of this User.

**Return type** list[*AccessPolicySummary*]

```
attribute_map = {'access_policies': 'accessPolicies', 'configurable': 'configurable'}
```

### **configurable**

Gets the configurable of this User. Indicates if this tenant is configurable, based on which UserGroup-Provider has been configured to manage it.

**Returns** The configurable of this User.

**Return type** bool

### **identifier**

Gets the identifier of this User. The computer-generated identifier of the tenant.

**Returns** The identifier of this User.

**Return type** str

### **identity**

Gets the identity of this User. The human-facing identity of the tenant. This can only be changed if the tenant is configurable.

**Returns** The identity of this User.

**Return type** str

### **resource\_permissions**

Gets the resource\_permissions of this User. A summary top-level resource access policies granted to this tenant.

**Returns** The resource\_permissions of this User.

**Return type** *ResourcePermissions*

### **revision**

Gets the revision of this User. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this User.

**Return type** RevisionInfo

```
swagger_types = {'access_policies': 'list[AccessPolicySummary]', 'configurable': 'bo
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
user_groups
    Gets the user_groups of this User. The groups to which the user belongs.
    Returns The user_groups of this User.
    Return type list[Tenant]
```

## nipyapi.registry.models.user\_group module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.user_group.UserGroup(identifier=None, identity=None,
                                                    configurable=None, re-
                                                    source_permissions=None,
                                                    access_policies=None,   revi-
                                                    sion=None, users=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

UserGroup - a model defined in Swagger

### access\_policies

Gets the access\_policies of this UserGroup. The access policies granted to this tenant.

**Returns** The access\_policies of this UserGroup.

**Return type** list[AccessPolicySummary]

```
attribute_map = {'access_policies': 'accessPolicies', 'configurable': 'configurable'
```

### configurable

Gets the configurable of this UserGroup. Indicates if this tenant is configurable, based on which User-GroupProvider has been configured to manage it.

**Returns** The configurable of this UserGroup.

**Return type** bool

### identifier

Gets the identifier of this UserGroup. The computer-generated identifier of the tenant.

**Returns** The identifier of this UserGroup.

**Return type** str

### identity

Gets the identity of this UserGroup. The human-facing identity of the tenant. This can only be changed if the tenant is configurable.

**Returns** The identity of this UserGroup.

**Return type** str

#### **resource\_permissions**

Gets the resource\_permissions of this UserGroup. A summary top-level resource access policies granted to this tenant.

**Returns** The resource\_permissions of this UserGroup.

**Return type** *ResourcePermissions*

#### **revision**

Gets the revision of this UserGroup. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this UserGroup.

**Return type** RevisionInfo

**swagger\_types** = {'access\_policies': 'list[AccessPolicySummary]', 'configurable': 'boolean'}

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **users**

Gets the users of this UserGroup. The users that belong to this user group. This can only be changed if this group is configurable.

**Returns** The users of this UserGroup.

**Return type** list[*Tenant*]

### **nipyapi.registry.models.versioned\_connection module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_connection.VersionedConnection (identifier=None,  
instance_identifier=None,  
name=None,  
comments=None,  
position=None,  
source=None,  
destination=None,  
label_index=None,  
z_index=None,  
selected_relationships=None,  
back_pressure_object_threshold=None,  
back_pressure_data_size_threshold=None,  
flow_file_expiration=None,  
priority=None,  
bends=None,  
load_balance_strategy=None,  
partitioning_attribute=None,  
load_balance_compression=None,  
ponent_type=None,  
group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedConnection - a model defined in Swagger

```
attribute_map = {'back_pressure_data_size_threshold': 'backPressureDataSizeThreshold'}
```

#### **back\_pressure\_data\_size\_threshold**

Gets the back\_pressure\_data\_size\_threshold of this VersionedConnection. The object data size threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The back\_pressure\_data\_size\_threshold of this VersionedConnection.

**Return type** str

#### **back\_pressure\_object\_threshold**

Gets the back\_pressure\_object\_threshold of this VersionedConnection. The object count threshold for determining when back pressure is applied. Updating this value is a passive change in the sense that it



won't impact whether existing files over the limit are affected but it does help feeder processors to stop pushing too much into this work queue.

**Returns** The `back_pressure_object_threshold` of this `VersionedConnection`.

**Return type** `int`

#### **bends**

Gets the bends of this `VersionedConnection`. The bend points on the connection.

**Returns** The bends of this `VersionedConnection`.

**Return type** `list[Position]`

#### **comments**

Gets the comments of this `VersionedConnection`. The user-supplied comments for the component

**Returns** The comments of this `VersionedConnection`.

**Return type** `str`

#### **component\_type**

Gets the `component_type` of this `VersionedConnection`.

**Returns** The `component_type` of this `VersionedConnection`.

**Return type** `str`

#### **destination**

Gets the destination of this `VersionedConnection`. The destination of the connection.

**Returns** The destination of this `VersionedConnection`.

**Return type** *ConnectableComponent*

#### **flow\_file\_expiration**

Gets the `flow_file_expiration` of this `VersionedConnection`. The amount of time a flow file may be in the flow before it will be automatically aged out of the flow. Once a flow file reaches this age it will be terminated from the flow the next time a processor attempts to start work on it.

**Returns** The `flow_file_expiration` of this `VersionedConnection`.

**Return type** `str`

#### **group\_identifier**

Gets the `group_identifier` of this `VersionedConnection`. The ID of the Process Group that this component belongs to

**Returns** The `group_identifier` of this `VersionedConnection`.

**Return type** `str`

#### **identifier**

Gets the identifier of this `VersionedConnection`. The component's unique identifier

**Returns** The identifier of this `VersionedConnection`.

**Return type** `str`

#### **instance\_identifier**

Gets the `instance_identifier` of this `VersionedConnection`. The instance ID of an existing component that is described by this `VersionedComponent`, or null if this is not mapped to an instantiated component

**Returns** The `instance_identifier` of this `VersionedConnection`.

**Return type** `str`

**label\_index**

Gets the label\_index of this VersionedConnection. The index of the bend point where to place the connection label.

**Returns** The label\_index of this VersionedConnection.

**Return type** int

**load\_balance\_compression**

Gets the load\_balance\_compression of this VersionedConnection. Whether or not compression should be used when transferring FlowFiles between nodes

**Returns** The load\_balance\_compression of this VersionedConnection.

**Return type** str

**load\_balance\_strategy**

Gets the load\_balance\_strategy of this VersionedConnection. The Strategy to use for load balancing data across the cluster, or null, if no Load Balance Strategy has been specified.

**Returns** The load\_balance\_strategy of this VersionedConnection.

**Return type** str

**name**

Gets the name of this VersionedConnection. The component's name

**Returns** The name of this VersionedConnection.

**Return type** str

**partitioning\_attribute**

Gets the partitioning\_attribute of this VersionedConnection. The attribute to use for partitioning data as it is load balanced across the cluster. If the Load Balance Strategy is configured to use PARTITION\_BY\_ATTRIBUTE, the value returned by this method is the name of the FlowFile Attribute that will be used to determine which node in the cluster should receive a given FlowFile. If the Load Balance Strategy is unset or is set to any other value, the Partitioning Attribute has no effect.

**Returns** The partitioning\_attribute of this VersionedConnection.

**Return type** str

**position**

Gets the position of this VersionedConnection. The component's position on the graph

**Returns** The position of this VersionedConnection.

**Return type** Position

**prioritizers**

Gets the prioritizers of this VersionedConnection. The comparators used to prioritize the queue.

**Returns** The prioritizers of this VersionedConnection.

**Return type** list[str]

**selected\_relationships**

Gets the selected\_relationships of this VersionedConnection. The selected relationship that comprise the connection.

**Returns** The selected\_relationships of this VersionedConnection.

**Return type** list[str]

**source**

Gets the source of this VersionedConnection. The source of the connection.

**Returns** The source of this VersionedConnection.

**Return type** *ConnectableComponent*

**swagger\_types** = {'back\_pressure\_data\_size\_threshold': 'str', 'back\_pressure\_object\_th

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**z\_index**

Gets the z\_index of this VersionedConnection. The z index of the connection.

**Returns** The z\_index of this VersionedConnection.

**Return type** int

### nipyapi.registry.models.versioned\_controller\_service module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_controller_service.VersionedControllerService (identified by its instance, name, comments, position, bundle, properties, controller, annotation_data, scheduled_bulletin_level, component_group)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedControllerService - a model defined in Swagger

**annotation\_data**

Gets the annotation\_data of this VersionedControllerService. The annotation for the controller service. This is how the custom UI relays configuration to the controller service.

**Returns** The annotation\_data of this VersionedControllerService.

**Return type** str

**attribute\_map** = {'annotation\_data': 'annotationData', 'bulletin\_level': 'bulletinLevel'}

**bulletin\_level**

Gets the bulletin\_level of this VersionedControllerService. The level at which the controller service will report bulletins.

**Returns** The bulletin\_level of this VersionedControllerService.

**Return type** str

**bundle**

Gets the bundle of this VersionedControllerService. Information about the bundle from which the component came

**Returns** The bundle of this VersionedControllerService.

**Return type** *Bundle*

**comments**

Gets the comments of this VersionedControllerService. The user-supplied comments for the component

**Returns** The comments of this VersionedControllerService.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedControllerService.

**Returns** The component\_type of this VersionedControllerService.

**Return type** str

**controller\_service\_apis**

Gets the controller\_service\_apis of this VersionedControllerService. Lists the APIs this Controller Service implements.

**Returns** The controller\_service\_apis of this VersionedControllerService.

**Return type** list[*ControllerServiceAPI*]

**group\_identifier**

Gets the group\_identifier of this VersionedControllerService. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedControllerService.

**Return type** str

**identifier**

Gets the identifier of this VersionedControllerService. The component's unique identifier

**Returns** The identifier of this VersionedControllerService.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedControllerService. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedControllerService.

**Return type** str

**name**

Gets the name of this VersionedControllerService. The component's name

**Returns** The name of this VersionedControllerService.

**Return type** str

**position**

Gets the position of this VersionedControllerService. The component's position on the graph

**Returns** The position of this VersionedControllerService.

**Return type** Position

**properties**

Gets the properties of this VersionedControllerService. The properties for the component. Properties whose value is not set will only contain the property name.

**Returns** The properties of this VersionedControllerService.

**Return type** dict(str, str)

**property\_descriptors**

Gets the property\_descriptors of this VersionedControllerService. The property descriptors for the component.

**Returns** The property\_descriptors of this VersionedControllerService.

**Return type** dict(str, *VersionedPropertyDescriptor*)

**scheduled\_state**

Gets the scheduled\_state of this VersionedControllerService. The ScheduledState denoting whether the Controller Service is ENABLED or DISABLED

**Returns** The scheduled\_state of this VersionedControllerService.

**Return type** str

**swagger\_types** = {'annotation\_data': 'str', 'bulletin\_level': 'str', 'bundle': 'Bund

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this VersionedControllerService. The type of the extension component

**Returns** The type of this VersionedControllerService.

**Return type** str

## nipyapi.registry.models.versioned\_flow module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_flow.VersionedFlow(link=None,      iden-
                                                         tifier=None,
                                                         name=None,      de-
                                                         scription=None,
                                                         bucket_identifier=None,
                                                         bucket_name=None,
                                                         cre-
                                                         ated_timestamp=None,
                                                         modi-
                                                         fied_timestamp=None,
                                                         type=None,      permis-
                                                         sions=None,      ver-
                                                         sion_count=None,
                                                         revision=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlow - a model defined in Swagger

```
attribute_map = {'bucket_identifier': 'bucketIdentifier', 'bucket_name': 'bucketName'}
```

**bucket\_identifier**

Gets the bucket\_identifier of this VersionedFlow. The identifier of the bucket this items belongs to. This cannot be changed after the item is created.

**Returns** The bucket\_identifier of this VersionedFlow.

**Return type** str

**bucket\_name**

Gets the bucket\_name of this VersionedFlow. The name of the bucket this items belongs to.

**Returns** The bucket\_name of this VersionedFlow.

**Return type** str

**created\_timestamp**

Gets the created\_timestamp of this VersionedFlow. The timestamp of when the item was created, as milliseconds since epoch.

**Returns** The created\_timestamp of this VersionedFlow.

**Return type** int

**description**

Gets the description of this VersionedFlow. A description of the item.

**Returns** The description of this VersionedFlow.

**Return type** str

**identifier**

Gets the identifier of this VersionedFlow. An ID to uniquely identify this object.

**Returns** The identifier of this VersionedFlow.

**Return type** str

**link**

Gets the link of this VersionedFlow. An WebLink to this entity.

**Returns** The link of this VersionedFlow.

**Return type** JaxbLink

**modified\_timestamp**

Gets the modified\_timestamp of this VersionedFlow. The timestamp of when the item was last modified, as milliseconds since epoch.

**Returns** The modified\_timestamp of this VersionedFlow.

**Return type** int

**name**

Gets the name of this VersionedFlow. The name of the item.

**Returns** The name of this VersionedFlow.

**Return type** str

**permissions**

Gets the permissions of this VersionedFlow. The access that the current user has to the bucket containing this item.

**Returns** The permissions of this VersionedFlow.

**Return type** *Permissions*

**revision**

Gets the revision of this VersionedFlow. The revision of this entity used for optimistic-locking during updates.

**Returns** The revision of this VersionedFlow.

**Return type** RevisionInfo

**swagger\_types** = {'bucket\_identifier': 'str', 'bucket\_name': 'str', 'created\_timestamp': 'str', 'flow\_id': 'str', 'latest': 'bool', 'revision': 'str', 'version': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this VersionedFlow. The type of item.

**Returns** The type of this VersionedFlow.

**Return type** str

**version\_count**

Gets the version\_count of this VersionedFlow. The number of versions of this flow.

**Returns** The version\_count of this VersionedFlow.

**Return type** int

## nipyapi.registry.models.versioned\_flow\_coordinates module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_flow_coordinates.VersionedFlowCoordinates(registry_url=  
    bucket_id=None, flow_id=None, latest=False, revision=None, version=None, lat=  
    est=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowCoordinates - a model defined in Swagger

**attribute\_map** = {'bucket\_id': 'bucketId', 'flow\_id': 'flowId', 'latest': 'latest', 'revision': 'revision', 'version': 'version', 'lat-est': 'lat-est'}

**bucket\_id**

Gets the bucket\_id of this VersionedFlowCoordinates. The UUID of the bucket that the flow resides in

**Returns** The bucket\_id of this VersionedFlowCoordinates.

**Return type** str



**flow\_id**

Gets the flow\_id of this VersionedFlowCoordinates. The UUID of the flow

**Returns** The flow\_id of this VersionedFlowCoordinates.

**Return type** str

**latest**

Gets the latest of this VersionedFlowCoordinates. Whether or not these coordinates point to the latest version of the flow

**Returns** The latest of this VersionedFlowCoordinates.

**Return type** bool

**registry\_url**

Gets the registry\_url of this VersionedFlowCoordinates. The URL of the Flow Registry that contains the flow

**Returns** The registry\_url of this VersionedFlowCoordinates.

**Return type** str

**swagger\_types** = {'bucket\_id': 'str', 'flow\_id': 'str', 'latest': 'bool', 'registry\_url': 'str'}

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**version**

Gets the version of this VersionedFlowCoordinates. The version of the flow

**Returns** The version of this VersionedFlowCoordinates.

**Return type** int

## nipyapi.registry.models.versioned\_flow\_snapshot module

### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_flow_snapshot.VersionedFlowSnapshot (snapshot_metadata=None,
                                                                              flow_contents=None,
                                                                              external_id=None,
                                                                              external_controller_service=None,
                                                                              parameter_names=None,
                                                                              external_id=None,
                                                                              flow_encoding_version=None,
                                                                              flow=None,
                                                                              bucket=None,
                                                                              latest=None)
```

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshot - a model defined in Swagger

```
attribute_map = {'bucket': 'bucket', 'external_controller_services': 'externalContro
```

**bucket**

Gets the bucket of this VersionedFlowSnapshot. The bucket where the flow is located

**Returns** The bucket of this VersionedFlowSnapshot.

**Return type** *Bucket*

**external\_controller\_services**

Gets the external\_controller\_services of this VersionedFlowSnapshot. The information about controller services that exist outside this versioned flow, but are referenced by components within the versioned flow.

**Returns** The external\_controller\_services of this VersionedFlowSnapshot.

**Return type** `dict(str, ExternalControllerServiceReference)`

**flow**

Gets the flow of this VersionedFlowSnapshot. The flow this snapshot is for

**Returns** The flow of this VersionedFlowSnapshot.

**Return type** *VersionedFlow*

**flow\_contents**

Gets the flow\_contents of this VersionedFlowSnapshot. The contents of the versioned flow

**Returns** The flow\_contents of this VersionedFlowSnapshot.

**Return type** *VersionedProcessGroup*

**flow\_encoding\_version**

Gets the flow\_encoding\_version of this VersionedFlowSnapshot. The optional encoding version of the flow contents.

**Returns** The flow\_encoding\_version of this VersionedFlowSnapshot.

**Return type** `str`

**latest**

Gets the latest of this VersionedFlowSnapshot.

**Returns** The latest of this VersionedFlowSnapshot.

**Return type** `bool`

**parameter\_contexts**

Gets the parameter\_contexts of this VersionedFlowSnapshot. The parameter contexts referenced by process groups in the flow contents. The mapping is from the name of the context to the context instance, and it is expected that any context in this map is referenced by at least one process group in this flow.

**Returns** The parameter\_contexts of this VersionedFlowSnapshot.

**Return type** `dict(str, VersionedParameterContext)`

**snapshot\_metadata**

Gets the snapshot\_metadata of this VersionedFlowSnapshot. The metadata for this snapshot

**Returns** The snapshot\_metadata of this VersionedFlowSnapshot.

**Return type** *VersionedFlowSnapshotMetadata*

```
swagger_types = {'bucket': 'Bucket', 'external_controller_services': 'dict(str, Ext
to_dict()
    Returns the model properties as a dict
to_str()
    Returns the string representation of the model
```

### nipyapi.registry.models.versioned\_flow\_snapshot\_metadata module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_flow_snapshot_metadata.VersionedFlowSnapshotMetadata
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFlowSnapshotMetadata - a model defined in Swagger

```
attribute_map = {'author': 'author', 'bucket_identifier': 'bucketIdentifier', 'comment'
```

**author**

Gets the author of this VersionedFlowSnapshotMetadata. The user that created this snapshot of the flow.

**Returns** The author of this VersionedFlowSnapshotMetadata.

**Return type** str

**bucket\_identifier**

Gets the bucket\_identifier of this VersionedFlowSnapshotMetadata. The identifier of the bucket this snapshot belongs to.

**Returns** The bucket\_identifier of this VersionedFlowSnapshotMetadata.

**Return type** str

**comments**

Gets the comments of this VersionedFlowSnapshotMetadata. The comments provided by the user when creating the snapshot.

**Returns** The comments of this VersionedFlowSnapshotMetadata.

**Return type** str

**flow\_identifier**

Gets the flow\_identifier of this VersionedFlowSnapshotMetadata. The identifier of the flow this snapshot belongs to.

**Returns** The flow\_identifier of this VersionedFlowSnapshotMetadata.

**Return type** str

**link**

Gets the link of this VersionedFlowSnapshotMetadata. An WebLink to this entity.

**Returns** The link of this VersionedFlowSnapshotMetadata.

**Return type** JaxbLink

**swagger\_types** = {'author': 'str', 'bucket\_identifier': 'str', 'comments': 'str', 'f

**timestamp**

Gets the timestamp of this VersionedFlowSnapshotMetadata. The timestamp when the flow was saved, as milliseconds since epoch.

**Returns** The timestamp of this VersionedFlowSnapshotMetadata.

**Return type** int

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**version**

Gets the version of this VersionedFlowSnapshotMetadata. The version of this snapshot of the flow.

**Returns** The version of this VersionedFlowSnapshotMetadata.

**Return type** int

## nipyapi.registry.models.versioned\_funnel module

### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_funnel.VersionedFunnel(identifier=None,  
                                                             in-  
                                                             stance_identifier=None,  
                                                             name=None,  
                                                             com-  
                                                             ments=None,  
                                                             position=None,  
                                                             compo-  
                                                             nent_type=None,  
                                                             group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedFunnel - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'component_type': 'componentType', 'group_i
```

**comments**

Gets the comments of this VersionedFunnel. The user-supplied comments for the component

**Returns** The comments of this VersionedFunnel.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedFunnel.

**Returns** The component\_type of this VersionedFunnel.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedFunnel. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedFunnel.

**Return type** str

**identifier**

Gets the identifier of this VersionedFunnel. The component's unique identifier

**Returns** The identifier of this VersionedFunnel.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedFunnel. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedFunnel.

**Return type** str

**name**

Gets the name of this VersionedFunnel. The component's name

**Returns** The name of this VersionedFunnel.

**Return type** str

**position**

Gets the position of this VersionedFunnel. The component's position on the graph

**Returns** The position of this VersionedFunnel.

**Return type** Position

```
swagger_types = {'comments': 'str', 'component_type': 'str', 'group_identifier': 's
```

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

### nipyapi.registry.models.versioned\_label module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_label.VersionedLabel (identifier=None, in-  
stance_identifier=None, name=None, com-  
ments=None, position=None, label=None, z_index=None, width=None, height=None, style=None, compo-  
nent_type=None, group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedLabel - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'component_type': 'componentType', 'group_i
```

#### **comments**

Gets the comments of this VersionedLabel. The user-supplied comments for the component

**Returns** The comments of this VersionedLabel.

**Return type** str

#### **component\_type**

Gets the component\_type of this VersionedLabel.

**Returns** The component\_type of this VersionedLabel.

**Return type** str

#### **group\_identifier**

Gets the group\_identifier of this VersionedLabel. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedLabel.

**Return type** str

#### **height**

Gets the height of this VersionedLabel. The height of the label in pixels when at a 1:1 scale.

**Returns** The height of this VersionedLabel.

**Return type** float

#### **identifier**

Gets the identifier of this VersionedLabel. The component's unique identifier

**Returns** The identifier of this VersionedLabel.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedLabel. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedLabel.

**Return type** str

**label**

Gets the label of this VersionedLabel. The text that appears in the label.

**Returns** The label of this VersionedLabel.

**Return type** str

**name**

Gets the name of this VersionedLabel. The component's name

**Returns** The name of this VersionedLabel.

**Return type** str

**position**

Gets the position of this VersionedLabel. The component's position on the graph

**Returns** The position of this VersionedLabel.

**Return type** Position

**style**

Gets the style of this VersionedLabel. The styles for this label (font-size : 12px, background-color : #eee, etc).

**Returns** The style of this VersionedLabel.

**Return type** dict(str, str)

**swagger\_types** = {'comments': 'str', 'component\_type': 'str', 'group\_identifier': 's

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**width**

Gets the width of this VersionedLabel. The width of the label in pixels when at a 1:1 scale.

**Returns** The width of this VersionedLabel.

**Return type** float

**z\_index**

Gets the z\_index of this VersionedLabel. The z index of the connection.

**Returns** The z\_index of this VersionedLabel.

**Return type** int

### nipyapi.registry.models.versioned\_port module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_port.VersionedPort (identifier=None, in-  
stance_identifier=None,  
name=None, com-  
ments=None,  
position=None,  
type=None, concur-  
rently_schedulable_task_count=None, sched-  
uled_state=None, al-  
low_remote_access=None,  
component_type=None,  
group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedPort - a model defined in Swagger

#### **allow\_remote\_access**

Gets the allow\_remote\_access of this VersionedPort. Whether or not this port allows remote access for site-to-site

**Returns** The allow\_remote\_access of this VersionedPort.

**Return type** bool

```
attribute_map = {'allow_remote_access': 'allowRemoteAccess', 'comments': 'comments',
```

#### **comments**

Gets the comments of this VersionedPort. The user-supplied comments for the component

**Returns** The comments of this VersionedPort.

**Return type** str

#### **component\_type**

Gets the component\_type of this VersionedPort.

**Returns** The component\_type of this VersionedPort.

**Return type** str

#### **concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this VersionedPort. The number of tasks that should be concurrently scheduled for the port.

**Returns** The concurrently\_schedulable\_task\_count of this VersionedPort.

**Return type** int

#### **group\_identifier**

Gets the group\_identifier of this VersionedPort. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedPort.

**Return type** str

#### **identifier**

Gets the identifier of this VersionedPort. The component's unique identifier



**Returns** The identifier of this VersionedPort.

**Return type** str

#### **instance\_identifier**

Gets the instance\_identifier of this VersionedPort. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedPort.

**Return type** str

#### **name**

Gets the name of this VersionedPort. The component's name

**Returns** The name of this VersionedPort.

**Return type** str

#### **position**

Gets the position of this VersionedPort. The component's position on the graph

**Returns** The position of this VersionedPort.

**Return type** Position

#### **scheduled\_state**

Gets the scheduled\_state of this VersionedPort. The scheduled state of the component

**Returns** The scheduled\_state of this VersionedPort.

**Return type** str

**swagger\_types** = {'allow\_remote\_access': 'bool', 'comments': 'str', 'component\_type':

#### **to\_dict()**

Returns the model properties as a dict

#### **to\_str()**

Returns the string representation of the model

#### **type**

Gets the type of this VersionedPort. The type of port.

**Returns** The type of this VersionedPort.

**Return type** str

## **nipyapi.registry.models.versioned\_process\_group module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_process_group.VersionedProcessGroup (identifier=None,
in-
stance_identifier=None,
name=None,
com-
ments=None,
po-
si-
tion=None,
pro-
cess_groups=None,
re-
mote_process_groups=None,
pro-
ces-
sors=None,
in-
put_ports=None,
out-
put_ports=None,
con-
nec-
tions=None,
la-
bels=None,
fun-
nels=None,
con-
troller_services=None,
ver-
sioned_flow_coordinator=None,
vari-
ables=None,
pa-
ram-
e-
ter_context_name=None,
de-
fault_flow_file_expiration=None,
de-
fault_back_pressure=None,
de-
fault_back_pressure_policy=None,
com-
po-
nent_type=None,
flow_file_concurrency=None,
flow_file_outbound_policy=None,
group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedProcessGroup - a model defined in Swagger

```
attribute_map = {'comments': 'comments', 'component_type': 'componentType', 'connect...
```

**comments**

Gets the comments of this VersionedProcessGroup. The user-supplied comments for the component

**Returns** The comments of this VersionedProcessGroup.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedProcessGroup.

**Returns** The component\_type of this VersionedProcessGroup.

**Return type** str

**connections**

Gets the connections of this VersionedProcessGroup. The Connections

**Returns** The connections of this VersionedProcessGroup.

**Return type** list[[VersionedConnection](#)]

**controller\_services**

Gets the controller\_services of this VersionedProcessGroup. The Controller Services

**Returns** The controller\_services of this VersionedProcessGroup.

**Return type** list[[VersionedControllerService](#)]

**default\_back\_pressure\_data\_size\_threshold**

Gets the default\_back\_pressure\_data\_size\_threshold of this VersionedProcessGroup. Default value used in this Process Group for the maximum data size of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_data\_size\_threshold of this VersionedProcessGroup.

**Return type** str

**default\_back\_pressure\_object\_threshold**

Gets the default\_back\_pressure\_object\_threshold of this VersionedProcessGroup. Default value used in this Process Group for the maximum number of objects that can be queued before back pressure is applied.

**Returns** The default\_back\_pressure\_object\_threshold of this VersionedProcessGroup.

**Return type** int

**default\_flow\_file\_expiration**

Gets the default\_flow\_file\_expiration of this VersionedProcessGroup. The default FlowFile Expiration for this Process Group.

**Returns** The default\_flow\_file\_expiration of this VersionedProcessGroup.

**Return type** str

**flow\_file\_concurrency**

Gets the flow\_file\_concurrency of this VersionedProcessGroup. The configured FlowFile Concurrency for the Process Group

**Returns** The flow\_file\_concurrency of this VersionedProcessGroup.

**Return type** str

**flow\_file\_outbound\_policy**

Gets the flow\_file\_outbound\_policy of this VersionedProcessGroup. The FlowFile Outbound Policy for the Process Group

**Returns** The flow\_file\_outbound\_policy of this VersionedProcessGroup.

**Return type** str

**funnels**

Gets the funnels of this VersionedProcessGroup. The Funnels

**Returns** The funnels of this VersionedProcessGroup.

**Return type** list[[VersionedFunnel](#)]

**group\_identifier**

Gets the group\_identifier of this VersionedProcessGroup. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedProcessGroup.

**Return type** str

**identifier**

Gets the identifier of this VersionedProcessGroup. The component's unique identifier

**Returns** The identifier of this VersionedProcessGroup.

**Return type** str

**input\_ports**

Gets the input\_ports of this VersionedProcessGroup. The Input Ports

**Returns** The input\_ports of this VersionedProcessGroup.

**Return type** list[[VersionedPort](#)]

**instance\_identifier**

Gets the instance\_identifier of this VersionedProcessGroup. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedProcessGroup.

**Return type** str

**labels**

Gets the labels of this VersionedProcessGroup. The Labels

**Returns** The labels of this VersionedProcessGroup.

**Return type** list[[VersionedLabel](#)]

**name**

Gets the name of this VersionedProcessGroup. The component's name

**Returns** The name of this VersionedProcessGroup.

**Return type** str

**output\_ports**

Gets the output\_ports of this VersionedProcessGroup. The Output Ports

**Returns** The output\_ports of this VersionedProcessGroup.

**Return type** list[[VersionedPort](#)]

**parameter\_context\_name**

Gets the parameter\_context\_name of this VersionedProcessGroup. The name of the parameter context used by this process group

**Returns** The parameter\_context\_name of this VersionedProcessGroup.

**Return type** str

**position**

Gets the position of this VersionedProcessGroup. The component's position on the graph

**Returns** The position of this VersionedProcessGroup.

**Return type** Position

**process\_groups**

Gets the process\_groups of this VersionedProcessGroup. The child Process Groups

**Returns** The process\_groups of this VersionedProcessGroup.

**Return type** list[*VersionedProcessGroup*]

**processors**

Gets the processors of this VersionedProcessGroup. The Processors

**Returns** The processors of this VersionedProcessGroup.

**Return type** list[*VersionedProcessor*]

**remote\_process\_groups**

Gets the remote\_process\_groups of this VersionedProcessGroup. The Remote Process Groups

**Returns** The remote\_process\_groups of this VersionedProcessGroup.

**Return type** list[*VersionedRemoteProcessGroup*]

**swagger\_types** = {'comments': 'str', 'component\_type': 'str', 'connections': 'list[V

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**variables**

Gets the variables of this VersionedProcessGroup. The Variables in the Variable Registry for this Process Group (not including any ancestor or descendant Process Groups)

**Returns** The variables of this VersionedProcessGroup.

**Return type** dict(str, str)

**versioned\_flow\_coordinates**

Gets the versioned\_flow\_coordinates of this VersionedProcessGroup. The coordinates where the remote flow is stored, or null if the Process Group is not directly under Version Control

**Returns** The versioned\_flow\_coordinates of this VersionedProcessGroup.

**Return type** *VersionedFlowCoordinates*

**nipyapi.registry.models.versioned\_processor module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_processor.VersionedProcessor (identifier=None,
                                                                    in-
                                                                    stance_identifier=None,
                                                                    name=None,
                                                                    com-
                                                                    ments=None,
                                                                    posi-
                                                                    tion=None,
                                                                    type=None,
                                                                    bun-
                                                                    dle=None,
                                                                    proper-
                                                                    ties=None,
                                                                    prop-
                                                                    erty_descriptors=None,
                                                                    style=None,
                                                                    an-
                                                                    nota-
                                                                    tion_data=None,
                                                                    schedul-
                                                                    ing_period=None,
                                                                    schedul-
                                                                    ing_strategy=None,
                                                                    execu-
                                                                    tion_node=None,
                                                                    penalty_duration=None,
                                                                    yield_duration=None,
                                                                    bul-
                                                                    letin_level=None,
                                                                    run_duration_millis=None,
                                                                    con-
                                                                    cur-
                                                                    rently_schedulable_task_count=None,
                                                                    auto_terminated_relationships=None,
                                                                    sched-
                                                                    uled_state=None,
                                                                    retry_count=None,
                                                                    re-
                                                                    tried_relationships=None,
                                                                    back-
                                                                    off_mechanism=None,
                                                                    max_backoff_period=None,
                                                                    compo-
                                                                    nent_type=None,
                                                                    group_identifier=None)
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedProcessor - a model defined in Swagger

#### **annotation\_data**

Gets the annotation\_data of this VersionedProcessor. The annotation data for the processor used to relay configuration between a custom UI and the procesosr.

**Returns** The annotation\_data of this VersionedProcessor.

**Return type** str

**attribute\_map** = {'annotation\_data': 'annotationData', 'auto\_terminated\_relationships'

**auto\_terminated\_relationships**

Gets the auto\_terminated\_relationships of this VersionedProcessor. The names of all relationships that cause a flow file to be terminated if the relationship is not connected elsewhere. This property differs from the 'isAutoTerminate' property of the RelationshipDTO in that the RelationshipDTO is meant to depict the current configuration, whereas this property can be set in a DTO when updating a Processor in order to change which Relationships should be auto-terminated.

**Returns** The auto\_terminated\_relationships of this VersionedProcessor.

**Return type** list[str]

**backoff\_mechanism**

Gets the backoff\_mechanism of this VersionedProcessor. Determines whether the FlowFile should be penalized or the processor should be yielded between retries.

**Returns** The backoff\_mechanism of this VersionedProcessor.

**Return type** str

**bulletin\_level**

Gets the bulletin\_level of this VersionedProcessor. The level at which the processor will report bulletins.

**Returns** The bulletin\_level of this VersionedProcessor.

**Return type** str

**bundle**

Gets the bundle of this VersionedProcessor. Information about the bundle from which the component came

**Returns** The bundle of this VersionedProcessor.

**Return type** *Bundle*

**comments**

Gets the comments of this VersionedProcessor. The user-supplied comments for the component

**Returns** The comments of this VersionedProcessor.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedProcessor.

**Returns** The component\_type of this VersionedProcessor.

**Return type** str

**concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this VersionedProcessor. The number of tasks that should be concurrently schedule for the processor. If the processor doesn't allow parallel processing then any positive input will be ignored.

**Returns** The concurrently\_schedulable\_task\_count of this VersionedProcessor.

**Return type** int

**execution\_node**

Gets the execution\_node of this VersionedProcessor. Indicates the node where the process will execute.

**Returns** The execution\_node of this VersionedProcessor.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedProcessor. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedProcessor.

**Return type** str

**identifier**

Gets the identifier of this VersionedProcessor. The component's unique identifier

**Returns** The identifier of this VersionedProcessor.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedProcessor. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedProcessor.

**Return type** str

**max\_backoff\_period**

Gets the max\_backoff\_period of this VersionedProcessor. Maximum amount of time to be waited during a retry period.

**Returns** The max\_backoff\_period of this VersionedProcessor.

**Return type** str

**name**

Gets the name of this VersionedProcessor. The component's name

**Returns** The name of this VersionedProcessor.

**Return type** str

**penalty\_duration**

Gets the penalty\_duration of this VersionedProcessor. The amount of time that is used when the process penalizes a flowfile.

**Returns** The penalty\_duration of this VersionedProcessor.

**Return type** str

**position**

Gets the position of this VersionedProcessor. The component's position on the graph

**Returns** The position of this VersionedProcessor.

**Return type** Position

**properties**

Gets the properties of this VersionedProcessor. The properties for the component. Properties whose value is not set will only contain the property name.

**Returns** The properties of this VersionedProcessor.

**Return type** dict(str, str)

**property\_descriptors**

Gets the property\_descriptors of this VersionedProcessor. The property descriptors for the component.

**Returns** The property\_descriptors of this VersionedProcessor.

**Return type** dict(str, *VersionedPropertyDescriptor*)



**retried\_relationships**

Gets the `retried_relationships` of this `VersionedProcessor`. All the relationships should be retried.

**Returns** The `retried_relationships` of this `VersionedProcessor`.

**Return type** `list[str]`

**retry\_count**

Gets the `retry_count` of this `VersionedProcessor`. Overall number of retries.

**Returns** The `retry_count` of this `VersionedProcessor`.

**Return type** `int`

**run\_duration\_millis**

Gets the `run_duration_millis` of this `VersionedProcessor`. The run duration for the processor in milliseconds.

**Returns** The `run_duration_millis` of this `VersionedProcessor`.

**Return type** `int`

**scheduled\_state**

Gets the `scheduled_state` of this `VersionedProcessor`. The scheduled state of the component

**Returns** The `scheduled_state` of this `VersionedProcessor`.

**Return type** `str`

**scheduling\_period**

Gets the `scheduling_period` of this `VersionedProcessor`. The frequency with which to schedule the processor. The format of the value will depend on the value of `schedulingStrategy`.

**Returns** The `scheduling_period` of this `VersionedProcessor`.

**Return type** `str`

**scheduling\_strategy**

Gets the `scheduling_strategy` of this `VersionedProcessor`. Indicates whether the processor should be scheduled to run in event or timer driven mode.

**Returns** The `scheduling_strategy` of this `VersionedProcessor`.

**Return type** `str`

**style**

Gets the `style` of this `VersionedProcessor`. Stylistic data for rendering in a UI

**Returns** The `style` of this `VersionedProcessor`.

**Return type** `dict(str, str)`

**swagger\_types** = {'annotation\_data': 'str', 'auto\_terminated\_relationships': 'list[str]

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**type**

Gets the type of this `VersionedProcessor`. The type of the extension component

**Returns** The type of this `VersionedProcessor`.

**Return type** `str`

**yield\_duration**

Gets the `yield_duration` of this `VersionedProcessor`. The amount of time that must elapse before this processor is scheduled again after yielding.

**Returns** The `yield_duration` of this `VersionedProcessor`.

**Return type** `str`

**nipyapi.registry.models.versioned\_property\_descriptor module**

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

**class** `nipyapi.registry.models.versioned_property_descriptor.VersionedPropertyDescriptor` (name

Bases: `object`

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

`VersionedPropertyDescriptor` - a model defined in Swagger

**attribute\_map** = {'display\_name': 'displayName', 'identifies\_controller\_service': 'id

**display\_name**

Gets the `display_name` of this `VersionedPropertyDescriptor`. The display name of the property

**Returns** The `display_name` of this `VersionedPropertyDescriptor`.

**Return type** `str`

**identifies\_controller\_service**

Gets the `identifies_controller_service` of this `VersionedPropertyDescriptor`. Whether or not the property provides the identifier of a Controller Service

**Returns** The `identifies_controller_service` of this `VersionedPropertyDescriptor`.

**Return type** `bool`

**name**

Gets the `name` of this `VersionedPropertyDescriptor`. The name of the property

**Returns** The name of this `VersionedPropertyDescriptor`.

**Return type** `str`

**resource\_definition**

Gets the `resource_definition` of this `VersionedPropertyDescriptor`. Returns the Resource Definition that defines which type(s) of resource(s) this property references, if any

**Return type** VersionedResourceDefinition

**sensitive**  
Gets the sensitive of this VersionedPropertyDescriptor. Whether or not the property is considered sensitive

**Returns** The sensitive of this VersionedPropertyDescriptor.

**Return type** bool

**swagger\_types** = {'display\_name': 'str', 'identifies\_controller\_service':

**to\_dict()**  
Returns the model properties as a dict

**to\_str()**  
Returns the string representation of the model

# Apache NiFi Registry REST API

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

Bases: object

## VersionedRemoteGroupPort - a model defined in Swagger

```
attribute_map = {'batch_size': 'batchSize', 'comments': 'comments', 'component_type':
```

**batch\_size**

Gets the batch\_size of this VersionedRemoteGroupPort. The batch settings for data transmission.

**Returns** The batch\_size of this VersionedRemoteGroupPort.

**Return type** *BatchSize*

**comments**

Gets the comments of this VersionedRemoteGroupPort. The user-supplied comments for the component

**Returns** The comments of this VersionedRemoteGroupPort.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedRemoteGroupPort.

**Returns** The component\_type of this VersionedRemoteGroupPort.

**Return type** str

**concurrently\_schedulable\_task\_count**

Gets the concurrently\_schedulable\_task\_count of this VersionedRemoteGroupPort. The number of task that may transmit flowfiles to the target port concurrently.

**Returns** The concurrently\_schedulable\_task\_count of this VersionedRemoteGroupPort.

**Return type** int

**group\_identifier**

Gets the group\_identifier of this VersionedRemoteGroupPort. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedRemoteGroupPort.

**Return type** str

**identifier**

Gets the identifier of this VersionedRemoteGroupPort. The component's unique identifier

**Returns** The identifier of this VersionedRemoteGroupPort.

**Return type** str

**instance\_identifier**

Gets the instance\_identifier of this VersionedRemoteGroupPort. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedRemoteGroupPort.

**Return type** str

**name**

Gets the name of this VersionedRemoteGroupPort. The component's name

**Returns** The name of this VersionedRemoteGroupPort.

**Return type** str

**position**

Gets the position of this VersionedRemoteGroupPort. The component's position on the graph

**Returns** The position of this VersionedRemoteGroupPort.

**Return type** Position

**remote\_group\_id**

Gets the remote\_group\_id of this VersionedRemoteGroupPort. The id of the remote process group that the port resides in.

**Returns** The remote\_group\_id of this VersionedRemoteGroupPort.

**Return type** str

**scheduled\_state**

Gets the scheduled\_state of this VersionedRemoteGroupPort. The scheduled state of the component

**Returns** The scheduled\_state of this VersionedRemoteGroupPort.

**Return type** str

**swagger\_types** = {'batch\_size': 'BatchSize', 'comments': 'str', 'component\_type': 's

**target\_id**

Gets the target\_id of this VersionedRemoteGroupPort. The ID of the port on the target NiFi instance

**Returns** The target\_id of this VersionedRemoteGroupPort.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**use\_compression**

Gets the use\_compression of this VersionedRemoteGroupPort. Whether the flowfiles are compressed when sent to the target port.

**Returns** The use\_compression of this VersionedRemoteGroupPort.

**Return type** bool

### nipyapi.registry.models.versioned\_remote\_process\_group module

#### Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.models.versioned_remote_process_group.VersionedRemoteProcessGroup (id
```

Bases: object

NOTE: This class is auto generated by the swagger code generator program. Do not edit the class manually.

VersionedRemoteProcessGroup - a model defined in Swagger

```
attribute_map = {'comments':  'comments', 'communications_timeout':  'communicationsTim
```

**comments**

Gets the comments of this VersionedRemoteProcessGroup. The user-supplied comments for the component

**Returns** The comments of this VersionedRemoteProcessGroup.

**Return type** str

**communications\_timeout**

Gets the communications\_timeout of this VersionedRemoteProcessGroup. The time period used for the timeout when communicating with the target.

**Returns** The communications\_timeout of this VersionedRemoteProcessGroup.

**Return type** str

**component\_type**

Gets the component\_type of this VersionedRemoteProcessGroup.

**Returns** The component\_type of this VersionedRemoteProcessGroup.

**Return type** str

**group\_identifier**

Gets the group\_identifier of this VersionedRemoteProcessGroup. The ID of the Process Group that this component belongs to

**Returns** The group\_identifier of this VersionedRemoteProcessGroup.

**Return type** str

**identifier**

Gets the identifier of this VersionedRemoteProcessGroup. The component's unique identifier

**Returns** The identifier of this VersionedRemoteProcessGroup.

**Return type** str

**input\_ports**

Gets the input\_ports of this VersionedRemoteProcessGroup. A Set of Input Ports that can be connected to, in order to send data to the remote NiFi instance

**Returns** The input\_ports of this VersionedRemoteProcessGroup.

**Return type** list[*VersionedRemoteGroupPort*]

**instance\_identifier**

Gets the instance\_identifier of this VersionedRemoteProcessGroup. The instance ID of an existing component that is described by this VersionedComponent, or null if this is not mapped to an instantiated component

**Returns** The instance\_identifier of this VersionedRemoteProcessGroup.

**Return type** str

**local\_network\_interface**

Gets the local\_network\_interface of this VersionedRemoteProcessGroup. The local network interface to send/receive data. If not specified, any local address is used. If clustered, all nodes must have an interface with this identifier.

**Returns** The local\_network\_interface of this VersionedRemoteProcessGroup.

**Return type** str

**name**

Gets the name of this VersionedRemoteProcessGroup. The component's name

**Returns** The name of this VersionedRemoteProcessGroup.

**Return type** str

**output\_ports**

Gets the output\_ports of this VersionedRemoteProcessGroup. A Set of Output Ports that can be connected to, in order to pull data from the remote NiFi instance

**Returns** The output\_ports of this VersionedRemoteProcessGroup.

**Return type** list[*VersionedRemoteGroupPort*]

**position**

Gets the position of this VersionedRemoteProcessGroup. The component's position on the graph

**Returns** The position of this VersionedRemoteProcessGroup.

**Return type** Position

**proxy\_host**

Gets the proxy\_host of this VersionedRemoteProcessGroup.

**Returns** The proxy\_host of this VersionedRemoteProcessGroup.

**Return type** str

**proxy\_port**

Gets the proxy\_port of this VersionedRemoteProcessGroup.

**Returns** The proxy\_port of this VersionedRemoteProcessGroup.

**Return type** int

**proxy\_user**

Gets the proxy\_user of this VersionedRemoteProcessGroup.

**Returns** The proxy\_user of this VersionedRemoteProcessGroup.

**Return type** str

**swagger\_types** = {'comments': 'str', 'communications\_timeout': 'str', 'component\_type'

**target\_uri**

Gets the target\_uri of this VersionedRemoteProcessGroup. [DEPRECATED] The target URI of the remote process group. If target uri is not set, but uris are set, then returns the first uri in the uris. If neither target uri nor uris are set, then returns null.

**Returns** The target\_uri of this VersionedRemoteProcessGroup.

**Return type** str

**target\_uris**

Gets the target\_uris of this VersionedRemoteProcessGroup. The target URIs of the remote process group. If target uris is not set but target uri is set, then returns the single target uri. If neither target uris nor target uri is set, then returns null.

**Returns** The target\_uris of this VersionedRemoteProcessGroup.

**Return type** str

**to\_dict()**

Returns the model properties as a dict

**to\_str()**

Returns the string representation of the model

**transport\_protocol**

Gets the transport\_protocol of this VersionedRemoteProcessGroup. The Transport Protocol that is used for Site-to-Site communications

**Returns** The transport\_protocol of this VersionedRemoteProcessGroup.

**Return type** str

**yield\_duration**

Gets the yield\_duration of this VersionedRemoteProcessGroup. When yielding, this amount of time must elapse before the remote process group is scheduled again.

**Returns** The yield\_duration of this VersionedRemoteProcessGroup.

**Return type** str



### 3.3.2.2 Submodules

#### 3.3.2.3 nipyapi.registry.api\_client module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
class nipyapi.registry.api_client.ApiClient (host=None, header_name=None,  
                                             header_value=None, cookie=None)
```

Bases: object

Generic API client for Swagger client library builds.

Swagger generic API client. This client handles the client- server communication, and is invariant across implementations. Specifics of the methods and models for each application are generated from the Swagger templates.

NOTE: This class is auto generated by the swagger code generator program. Ref: <https://github.com/swagger-api/swagger-codegen> Do not edit the class manually.

##### Parameters

- **host** – The base path for the server to call.
- **header\_name** – a header to pass when making calls to the API.
- **header\_value** – a header value to pass when making calls to the API.

Constructor of the class.

```
NATIVE_TYPES_MAPPING = {'bool': <class 'bool'>, 'date': <class 'datetime.date'>, 'da  
PRIMITIVE_TYPES = (<class 'float'>, <class 'bool'>, <class 'bytes'>, <class 'str'>, <c
```

```
call_api (resource_path, method, path_params=None, query_params=None, header_params=None,  
          body=None, post_params=None, files=None, response_type=None, auth_settings=None,  
          callback=None, _return_http_data_only=None, collection_formats=None,  
          _preload_content=True, _request_timeout=None)
```

Makes the HTTP request (synchronous) and return the deserialized data. To make an async request, define a function for callback.

##### Parameters

- **resource\_path** – Path to method endpoint.
- **method** – Method to call.
- **path\_params** – Path parameters in the url.
- **query\_params** – Query parameters in the url.
- **header\_params** – Header parameters to be placed in the request header.
- **body** – Request body.
- **dict** (*files*) – Request post form parameters, for *application/x-www-form-urlencoded*, *multipart/form-data*.
- **list** (*auth\_settings*) – Auth Settings names for the request.
- **response** – Response data type.
- **dict** – key -> filename, value -> filepath, for *multipart/form-data*.

- **function** (*callback*) – Callback function for asynchronous request. If provide this parameter, the request will be called asynchronously.
- **\_return\_http\_data\_only** – response data without head status code and headers
- **collection\_formats** – dict of collection formats for path, query, header, and post parameters.
- **\_preload\_content** – if False, the urllib3.HTTPResponse object will be returned without reading/decoding response data. Default is True.
- **\_request\_timeout** – timeout setting for this request. If one number provided, it will be total request timeout. It can also be a pair (tuple) of (connection, read) timeouts.

**Returns** If provide parameter callback, the request will be called asynchronously. The method will return the request thread. If parameter callback is None, then the method will return the response directly.

**deserialize** (*response, response\_type*)

Deserializes response into an object.

**Parameters**

- **response** – RESTResponse object to be deserialized.
- **response\_type** – class literal for deserialized object, or string of class name.

**Returns** deserialized object.

**deserialize\_model** (*data, klass*)

Deserializes list or dict to model.

**Parameters**

- **data** – dict, list.
- **klass** – class literal.

**Returns** model object.

**parameters\_to\_tuples** (*params, collection\_formats*)

Get parameters as list of tuples, formatting collections.

**Parameters**

- **params** – Parameters as dict or list of two-tuples
- **collection\_formats** (*dict*) – Parameter collection formats

**Returns** Parameters as list of tuples, collections formatted

**prepare\_post\_parameters** (*post\_params=None, files=None*)

Builds form parameters.

**Parameters**

- **post\_params** – Normal form parameters.
- **files** – File parameters.

**Returns** Form parameters with files.

**request** (*method, url, query\_params=None, headers=None, post\_params=None, body=None, \_preload\_content=True, \_request\_timeout=None*)

Makes the HTTP request using RESTClient.

**sanitize\_for\_serialization** (*obj*)

Builds a JSON POST object.

If *obj* is None, return None. If *obj* is str, int, long, float, bool, return directly. If *obj* is datetime.datetime, datetime.date

convert to string in iso8601 format.

If *obj* is list, sanitize each element in the list. If *obj* is dict, return the dict. If *obj* is swagger model, return the properties dict.

**Parameters** *obj* – The data to serialize.

**Returns** The serialized form of data.

**select\_header\_accept** (*accepts*)

Returns *Accept* based on an array of accepts provided.

**Parameters** *accepts* – List of headers.

**Returns** Accept (e.g. application/json).

**select\_header\_content\_type** (*content\_types*)

Returns *Content-Type* based on an array of content\_types provided.

**Parameters** *content\_types* – List of content-types.

**Returns** Content-Type (e.g. application/json).

**set\_default\_header** (*header\_name, header\_value*)**update\_params\_for\_auth** (*headers, querys, auth\_settings*)

Updates header and query params based on authentication setting.

**Parameters**

- **headers** – Header parameters dict to be updated.
- **querys** – Query parameters tuple list to be updated.
- **auth\_settings** – Authentication setting identifiers list.

**user\_agent**

Gets user agent.

### 3.3.2.4 nipyapi.registry.configuration module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

`nipyapi.registry.configuration.Configuration()`

`nipyapi.registry.configuration.singleton(cls, *args, **kw)`

### 3.3.2.5 nipyapi.registry.rest module

Apache NiFi Registry REST API

The REST API provides an interface to a registry with operations for saving, versioning, reading NiFi flows and components.

OpenAPI spec version: 1.17.0 Contact: [dev@nifi.apache.org](mailto:dev@nifi.apache.org) Generated by: <https://github.com/swagger-api/swagger-codegen.git>

```
exception nipyapi.registry.rest.ApiException (status=None, reason=None,
                                              http_resp=None)
```

Bases: Exception

```
class nipyapi.registry.rest.RESTClientObject (pools_size=4, maxsize=4)
```

Bases: object

```
DELETE (url, headers=None, query_params=None, body=None, _preload_content=True, _request_timeout=None)
```

```
GET (url, headers=None, query_params=None, _preload_content=True, _request_timeout=None)
```

```
HEAD (url, headers=None, query_params=None, _preload_content=True, _request_timeout=None)
```

```
OPTIONS (url, headers=None, query_params=None, post_params=None, body=None,
          _preload_content=True, _request_timeout=None)
```

```
PATCH (url, headers=None, query_params=None, post_params=None, body=None,
         _preload_content=True, _request_timeout=None)
```

```
POST (url, headers=None, query_params=None, post_params=None, body=None,
       _preload_content=True, _request_timeout=None)
```

```
PUT (url, headers=None, query_params=None, post_params=None, body=None,
     _preload_content=True, _request_timeout=None)
```

```
request (method, url, query_params=None, headers=None, body=None, post_params=None,
         _preload_content=True, _request_timeout=None)
```

#### Parameters

- **method** – http request method
- **url** – http request url
- **query\_params** – query parameters in the url
- **headers** – http request headers
- **body** – request json body, for *application/json*
- **post\_params** – request post parameters, *application/x-www-form-urlencoded* and *multipart/form-data*
- **\_preload\_content** – if False, the urllib3.HTTPResponse object will be returned without reading/decoding response data. Default is True.
- **\_request\_timeout** – timeout setting for this request. If one number provided, it will be total request timeout. It can also be a pair (tuple) of (connection, read) timeouts.

```
class nipyapi.registry.rest.RESTResponse (resp)
```

Bases: io.IOBase

```
getheader (name, default=None)
```

Returns a given response header.

```
getheaders ()
```

Returns a dictionary of the response headers.

## CHAPTER 4

---

### ToDo

---

- Bring accross features from <https://github.com/pvillard31/nifi-api-client-python>
- Bring across features from <https://github.com/jdye64/nifi-shell>
- Look into how sensitive properties are handled on template import
- Add a depth limit to the recursion on the canvas flow fetcher
- Add specific ‘secure mode’ switch which allows commands to run with simple defaults
- <https://community.hortonworks.com/articles/56849/automate-deployment-of-hdf-20-clusters-using-ambar.html>
- <https://github.com/hayanige/docker-nifi-cluster>
- Set enforcement of ProcessGroup best practice like unique template names
- Create Ansible wrappers for executing against Yaml
- Create more deterministic deploy/reset/kill controls for Demo/Test fixtures
- Setup regression testing to handle forward version logic

Please see the [issue](#) register for more information on current development.



Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 5.1 Types of Contributions

### 5.1.1 Report Bugs

Report bugs at <https://github.com/Chaffelson/nipyapi/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

### 5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

### 5.1.4 Write Documentation

Nipyapi could always use more documentation, whether as part of the official Nipyapi docs, in docstrings, or even on the web in blog posts, articles, and such.

### 5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/Chaffelson/nipyapi/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 5.2 Get Started!

Ready to contribute? Here's how to set up *nipyapi* for local development.

1. Fork the *nipyapi* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:Chaffelson/nipyapi.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv nipyapi
$ cd nipyapi/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. You may want to leverage the provided Docker configuration for testing and development

- Install the latest version of Docker
- Use the provided Docker Compose configuration in `./resources/docker/latest` and run the tests:

```
$ cd resources/docker/latest
$ docker-compose up -d
$ cd ../../../../
$ tox
$ cd resources/docker/latest
$ docker-compose stop
```

6. You may also want to interactively test your code leveraging the convenience console in the demo package:

```
$ python
> from nipyapi.demo.console import *
```



7. When you're done making changes, check that your changes pass the tests, including testing other Python versions, with tox:

```
$ tox
```

8. Commit your changes and push your branch to GitHub:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

9. Submit a pull request through the GitHub website.

## 5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7 and 3.6, and for PyPy. Check [https://travis-ci.org/Chaffelson/nipyapi/pull\\_requests](https://travis-ci.org/Chaffelson/nipyapi/pull_requests) and make sure that the tests pass for all supported Python versions.
4. Pull requests should be created against the 'next' branch for new features, or 'master' for critical patches to current functionality.



---

## Development Notes

---

A collection point for information about the development process for future collaborators

### 6.1 Decision Points

- Using Swagger 2.0 instead of OpenAPI3.0 as it (currently as of Aug2017) has wider adoption and completed codegen tools
- We use Google style Docstrings to better enable Sphinx to produce nicely readable documentation

### 6.2 Testing Notes

When running tests on new code, you are advised to run ‘test\_default’ first, then ‘test\_regression’, then finally ‘test\_security’. Because of the way errors are propagated you may have code failures which cause a teardown which then fails because of security controls, which can then obscure the original error.

### 6.3 Docker Test Environment

There is an Apache NiFi image available on Dockerhub:

```
docker pull apache/nifi:latest
```

There are a couple of configuration files for launching various Docker environment configurations in ./test\_env\_config for convenience.

### 6.4 Remote Testing on Centos7

Deploy a 4x16 or better on EC2 running Centos 7.5 or better, ssh in as root:

```
yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
yum update -y
yum install -y centos-release-scl yum-utils device-mapper-persistent-data lvm2
yum install -y rh-python36 docker-ce docker-ce-cli containerd.io
systemctl start docker
scl enable rh-python36 bash
sudo curl -L "https://github.com/docker/compose/releases/download/1.25.0/docker-
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

Set up remote execution environment to this server from your IDE, such as PyCharm. Python3 will be in a path like /opt/rh/rh-python36/root/usr/bin/python These commands are conveniently presented in /resources/test\_setup/setup\_centos7.sh

You will then want to open up /home/centos/tmp/<pycharmprojectname>/resources/docker/tox-full and run:

```
docker-compose pull
docker-compose up -d
```

## 6.5 Testing on OSX

There is a known issue with testing newer versions of Python on OSX. You may receive an error reporting [SSL: CERTIFICATE\_VERIFY\_FAILED] when trying to install packages from Pypi

You can fix this by running the following commands:

```
export PIP_REQUIRE_VIRTUALENV=false
/Applications/Python\ 3.6/Install\ Certificates.command
```

## 6.6 Generate Swagger Client

The NiFi and NiFi Registry REST API clients are generated using swagger-codegen, which is available via a variety of methods:

- the package manager for your OS
- github: <https://github.com/swagger-api/swagger-codegen>
- maven: <http://central.maven.org/maven2/io/swagger/swagger-codegen-cli/2.3.1/swagger-codegen-cli-2.3.1.jar>
- pre-built Docker images on DockerHub (<https://hub.docker.com/r/swaggerapi/swagger-codegen-cli/>)

In the examples below, we'll use Homebrew for macOS:

```
brew install swagger-codegen
```

### 6.6.1 NiFi Swagger Client

1. build relevant version of NiFi from source
2. use swagger-codegen to generate the Python client:

```
mkdir -p ~/tmp && \
echo '{ "packageName": "nifi" }' > ~/tmp/swagger-nifi-python-config.json && \
rm -rf ~/tmp/nifi-python-client && \
swagger-codegen generate \
  --lang python \
  --config swagger-nifi-python-config.json \
  --api-package apis \
  --model-package models \
  --template-dir /path/to/nipyapi/swagger_templates \
  --input-spec /path/to/nifi/nifi-nar-bundles/nifi-framework-bundle/nifi-
↪framework/nifi-web/nifi-web-api/target/swagger-ui/swagger.json \
  --output ~/tmp/nifi-python-client
```

3. replace the embedded clients:

```
rm -rf /path/to/nipyapi/nipyapi/nifi && cp -rf ~/tmp/nifi-python-client/nifi /
↪path/to/nipyapi/nipyapi/nifi
```

4. review the changes and submit a PR!

## 6.6.2 NiFi Registry Swagger Client

1. Fetch the definition from a running Registry instance at URI: /nifi-registry-api/swagger/swagger.json
2. use swagger-codegen to generate the Python client:

```
mkdir -p ~/tmp && \
echo '{ "packageName": "registry" }' > ~/tmp/swagger-registry-python-config.json &
↪& \
rm -rf ~/tmp/nifi-registry-python-client && \
swagger-codegen generate \
  --lang python \
  --config swagger-registry-python-config.json \
  --api-package apis \
  --model-package models \
  --template-dir /path/to/nipyapi/swagger_templates \
  --input-spec /path/to/nifi-registry/nifi-registry-web-api/target/swagger-ui/
↪swagger.json \
  --output ~/tmp/nifi-registry-python-client
```

3. replace the embedded clients:

```
rm -r /path/to/nipyapi/nipyapi/registry && cp -rf /tmp/nifi-registry-python-
↪client/swagger_client /path/to/nipyapi/nipyapi/registry
```

4. review the changes and submit a PR!

## 6.7 Release Process

This assumes you have virtualenvwrapper, git, and appropriate python versions installed, as well as the necessary test environment:

- update History.rst
- check setup.py

- check requirements.txt and requirements\_dev.txt
- Commit all changes
- in bash:

```
cd ProjectDir
source ./my_virtualenv/bin/activate
bumpversion patch|minor|major
python setup.py develop
tox
python setup.py test
python setup.py build_sphinx
# check docs in build/sphinx/html/index.html
python setup.py sdist bdist_wheel
mktmpenv # or pyenv virtualenvwrapper mktmpenv if using pyenv
pip install path/to/nipyapi-0.3.1-py2.py3-none-any.whl # for example
# Run appropriate tests, such as usage tests etc.
deactivate
Push changes to Github
Check build on TravisCI
Check dockerhub automated build
# You may have to reactivate your original virtualenv
twine upload dist/*
# You may get a file exists error, check you're not trying to reupload an_
↪existing version
git push --tags
```

- check build in TravisCI
- check docs on ReadTheDocs
- check release published on Github and PyPi

### 7.1 Development Lead

- Daniel Chaffelson <[chaffelson@gmail.com](mailto:chaffelson@gmail.com)>

### 7.2 Contributors

Kevin Doran <[kdoran@apache.org](mailto:kdoran@apache.org)>

### 7.3 Shout Outs

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template

Inspired by the equivalent Java client maintained over at [hermannpencole/nifi-config](#)

The swagger 2.0 compliant client auto-generated using the [Swagger Codegen](#) project, and then cleaned / bugfixed by the authors

Props to the NiFi-dev and NiFi-user mailing list members over at Apache for all the assistance and kindnesses.





### 8.1 0.19.1 (2022-08-08)

Updated client for NiFi & Registry 1.17.0 release

- Update docker resources to the latest 1.17 container image
- Fix issue in docker volume mounts where certificates were not readable in new versions of Docker
- Fix issue in `test_get_processor_type` where new Twitter processor broke existing logic assumptions
- Update readme to note issues with Python 3.10 and removing reference to python 3.5 testing

### 8.2 0.19.0 (2022-05-13)

Updated client for NiFi & Registry 1.16.0 release

- Modify `utils.check_version` to accept a default version to check against, and remove nested error when version check fails as it is overzealous
- Update test file paths in `test_utils` in case someone runs tests as root which actually can write to fake devices and therefore break the tests
- Added pylint exclusions for known but unimportant complaints
- Added workaround for NiFi 9470 by providing default `inherited_parameter_contexts` per advice from Chris Sampson in NiPyAPI Issue #305
- Updated NiFi and Registry versions in Docker compose configs to use latest 1.16.1 builds

## 8.3 0.18.0 (2021-11-29)

Updated client for NiFi & Registry 1.15.0 release

- Moved testing to secured single user by default for NiFi
- Consolidated clients to the 1.15.0 release version for both NiFi and Registry
- Updated logic to support changed working modes in tests
- Corrected minor functional issues for 1.15 in login handling, SSL certs, Parameter Updates and Testing
- Backwards compatibility and Regression tested as requiring no breaking changes

## 8.4 0.17.1 (2021-10-21)

Added gzip as default request header (PR from rsaggino)

Update ruamel.yaml to 0.17.16

Update file read/write to explicitly handle encoding, default to UTF8, configurable in nipyapi.config

Linting and style changes to update to Python 3.9 standards, except where it would break backwards compatibility to Python 2.7

Test support for Amazon Linux 2, mostly better error handling for filesystem responses

## 8.5 0.17.0 (2021-10-13)

Updated NiFi version and client to 1.13.2

Updated NiFi-Registry version and client to 0.8.0

- Note that these are the last versions where NiFi and NiFi-Registry are separate codebases

## 8.6 0.16.3 (2021-10-11)

Removed force reset of configuration.password and configuration.username to empty string. This was not increasing security, and was causing unexpected errors for users connecting to multiple services in a single script.

Add greedy control to versioning.get\_registry\_bucket and versioning.get\_flow\_in\_bucket to avoid undesirable partial string match.

- Update readme to reflect switch from 'master' branch naming to 'main'.
- Update tox to pin testing to Python 3.8, as Python 3.9 is producing unexpected and unrelated SSL failures
- Minor lint formatting improvements

## 8.7 0.16.2 (2021-02-10)

NOTE: If you are using secured Registry, this release will enforce access controls for the swagger interface which is used to determine which version of Registry is connected in order to correctly provide features - you may have to

update your authorizations

- Update requirements.txt to unpin future and lxml
- Update lxml to 4.6.2 or newer to resolve vulnerability
- Pin watchdog to <1.0.0 per their docs to maintain Python2.7 compatibility
- Revert 0.14.3 changes to Authentication handling which introduced basicAuth support but resulted in some NiFi connections appearing incorrectly as Anonymous
- Added simpler basicAuth control to force it via a config switch without changing tokenAuth and other Authorization header behavior during normal usage
- nipyapi.config.global\_force\_basic\_auth is now available for use for this purpose
- Secured Registry users will now require the authorization policy to retrieve the swagger so we may use it to validate which version of
- Registry is in use for feature enablement
- Moved all Security controls in config.py to a common area at the foot of the file
- Removed auth\_type from security.service\_login as it is now redundant
- Added controls to handle certificate checking behavior which has become more strict in recently versions of Python3, ssl\_verify and check\_hostname are now handled
- security.set\_service\_auth\_token now has an explicit flag for ssl host checking as well
- Fix oversight where improved model serialisation logic was not correctly applied to Registry
- Removed unused parameter refresh from parameters.update\_parameter\_context
- Reduced unnecessary complexity in utils.dump with no change in functionality
- Updated client gen mustache templates to reflect refactored security and api client code
- Minor linting and docstring and codestyle improvements
- Set pyUp to ignore Watchdog as it must stay between versions to satisfy py2 and py3 compatibility
- If Client is not instantiated, optimistically instantiate for version checking
- add socks proxy support

## 8.8 0.16.3 (2021-10-11)

Removed force reset of configuration.password and configuration.username to empty string. This was not increasing security, and was causing unexpected errors for users connecting to multiple services in a single script.

Add greedy control to versioning.get\_registry\_bucket and versioning.get\_flow\_in\_bucket to avoid undesirable partial string match.

- Update readme to reflect switch from ‘master’ branch naming to ‘main’.
- Update tox to pin testing to Python 3.8, as Python 3.9 is producing unexpected and unrelated SSL failures
- Minor lint formatting improvements

## 8.9 0.16.2 (2021-02-10)

NOTE: If you are using secured Registry, this release will enforce access controls for the swagger interface which is used to determine which version of Registry is connected in order to correctly provide features - you may have to update your authorizations

- Update requirements.txt to unpin future and lxml
- Update lxml to 4.6.2 or newer to resolve vulnerability
- Pin watchdog to <1.0.0 per their docs to maintain Python2.7 compatibility
- Revert 0.14.3 changes to Authentication handling which introduced basicAuth support but resulted in some NiFi connections appearing incorrectly as Anonymous
- Added simpler basicAuth control to force it via a config switch without changing tokenAuth and other Authorization header behavior during normal usage
- nipyapi.config.global\_force\_basic\_auth is now available for use for this purpose
- Secured Registry users will now require the authorization policy to retrieve the swagger so we may use it to validate which version of
- Registry is in use for feature enablement
- Moved all Security controls in config.py to a common area at the foot of the file
- Removed auth\_type from security.service\_login as it is now redundant
- Added controls to handle certificate checking behavior which has become more strict in recently versions of Python3, ssl\_verify and check\_hostname are now handled
- security.set\_service\_auth\_token now has an explicit flag for ssl host checking as well
- Fix oversight where improved model serialisation logic was not correctly applied to Registry
- Removed unused parameter refresh from parameters.update\_parameter\_context
- Reduced unnecessary complexity in utils.dump with no change in functionality
- Updated client gen mustache templates to reflect refactored security and api client code
- Minor linting and docstring and codestyle improvements
- Set pyUp to ignore Watchdog as it must stay between versions to satisfy py2 and py3 compatibility
- If Client is not instantiated, optimistically instantiate for version checking
- add socks proxy support

## 8.10 0.15.0 (2020-11-06)

Updated NiFi client and helpers to 1.12.1, Registry client to 0.7.0

Release to include new fixes and features in baseline, work continues on improving different Authentication methods

- Added new Parameter contexts API to docs
- Resolved bug where funnel position did not honour requested location (thanks @geocali)
- Fixed issue where users expected exact search by default but some functions were silently using greedy search (thanks @razdob15)

- Change `deploy_template` to use floats for deployment instead of int (thanks @bgeisberger)
- Fixed creation of empty user groups (thanks @razdob15)

## 8.11 0.14.0 (2019-11-06)

Updated NiFi client and helpers to 1.10.0

## 8.12 0.13.3 (2019-10-09)

Updated NiFi-Registry client for 0.5.0

Several Issues closed as bugfixes

Many canvas operations sped-up through refactoring of recursive code to fast iterators

## 8.13 0.13.0 (2019-04-22)

Updated NiFi client for 1.9.1

Major rework of `security.py` to handle TLS and BasicAuth scenarios

Major rework for `test_security.py` to cover Issues and common use cases

Update `'set_endpoint'` to easily handle TLS and BasicAuth scenarios if https is set

- Add default BasicAuth params to config
- Add default `'safe chars'` to config for URL encoding bypass where `'/'` is in a string
- Add `'bypass_slash_encoding'` to `utils.py` to simplify conditionally allowing `'/'` in a string
- Update Docker compose files for Secure and tox-full environments to latest NiFi versions
- Add global test controls to top of `confest` for default, security, and regression test modes
- Add fixtures to `confest` for user and usergroup testing in secure scenarios
- Update fixtures to better handle mixed secure and insecure test environments

## 8.14 0.12.0 (2018-12-20)

Updated NiFi client for 1.8.0

Updated NiFi-Registry client for 0.3.0

Added Controller Service Management (experimental)

Added Connections Management (experimental)

New Project Logo! Kindly provided by KDoran

Fixed several bugs around how the special root Process Group is handled when listing all Project Groups for various methods

- Various backwards compatibility improvements for handling calls going back to NiFi-1.1.2
- Various speedups for NiFi-1.7+ using descendants functionality to recurse the canvas

- Ability for various methods to specify a Process Group to use as the parent instead of always using root
- Better username/password handling in security.py and config.py
- Support for global ssl\_verify squashing in config.py
- Added swagger for 1.8.0 to project resources against potential future validation requirements
- Added versioned deployment convenience functions for finding sensitive and invalid processors, should make it easier to update properties when importing to a new canvas
- Added summary options to several calls to return simple objects suitable for quick processing rather than full objects that need to be parsed
- Added utils.infer\_object\_label\_from\_class to make it easier to create connections between objects
- Updated compound methods like delete\_process\_group to also handle connections and controllers elegantly if requested
- Various codestyle and testing improvements

## **8.15 0.11.0 (2018-10-12)**

Added steps to fdlc demo to show sensitive and invalid processor testing and behavior during deployment

Added list\_sensitive\_processors and list\_invalid\_processors to nipyapi.canvas

Added simple caching capability for certain calls to nipyapi.config

Added placeholder tests for new functionality against next refactoring and integration run

Missing assertion test in get\_process\_group\_status

deprecated use of tests\_require setup.py as current best-practice

Update ruamel.yaml to support Python 3.7 with passing tests

Added test for docker image already present to avoid excessive downloading

Added option to recurse from a given pg\_id, rather than always from root, to several canvas functions

Added default verify\_ssl and ssl error squashing to config for user convenience

Added filter option to specify whether exact or greedy matching should be used, still greedy by default

Added hard logout when changing endpoint to ensure tokens are refreshed

Updated tests

updating travis to build all branches

Fix travis for Python 3.7 testing support

Fix edge case in delete process group where templates stop the revision from being refreshed

Fixed test case to decode string correctly in old python versions

Fixed race condition in test where not all processors started before test executes

bugfix for missing status value in Processor DTO

Updating pylint to ignore import errors on standard packages

added logging to docker image control

Bump version: 0.10.3 → 0.11.0

Install requirements reset

## **8.16 0.10.3 (2018-08-28)**

Minor bugfix for versioning/deploy\_flow\_version to resolve additional edge case for version number type

## 8.17 0.10.2 (2018-08-27)

BugFix for Issue #66 in security/get\_access\_policy\_for\_resource where NiFi Api is not expecting a resource\_id to be submitted

## 8.18 0.10.1 (2018-08-21)

Minor bugfix for versioning/deploy\_flow\_version where version number should be a str instead of int

## 8.19 0.10.0 (2018-08-03)

Updated NiFi client for 1.7.1 release

Updated NiFi-Registry client for 0.2.0 release

### Key Changes

- Reworked NiFi-Registry pytest setup to support multiple versions
- Changed schedule\_processor to use component. rather than status. tests as they are more reliable
- Switched Docker configs to use explicit versions instead of latest for more consistent behavior across environments

### Version Changes

- Deprecated testing against NiFi-1.5.0 due to host headers issue - recommend users to upgrade to at least NiFi-1.6.0
- Deprecated testing against NiFi-1.4.0 as superfluous
- Added testing for NiFi-1.7.1 and NiFi-Registry-0.2.0

## 8.20 0.9.1 (2018-05-18)

Updated Demos for 0.9 release

### New Features

- Added a new demo for Flow Development LifeCycle which illustrates the steps a user might automate to promote Versioned Flows between NiFi environments
- Check out nipyapi.demo.fdlc to see more details

## 8.21 0.9.0 (2018-05-16)

Updated NiFi client to 1.6.0 release

### Potentially Breaking Changes

*Users should check the updated documentation and ensure their tests pass as expected*

- Several NiFi client API calls were inconsistently CamelCase'd and have been renamed in the upstream NiFi release, I have honoured those changes in this release. If you use them please check your function names if you get an error

### New Features

- Added functionality to Deploy a versioned flow to the canvas. This was an oversight from the 0.8.0 release. Function is creatively named `./versioning/deploy_flow_version`

### Other Notes

- Updated the Issue Template to also ask how urgent the problem is so we can prioritise work
- Where possible we have switched to using the Apache maintained Docker containers rather than our own, there should be no impact to this unless you were relying on some edge part of our test compose files

## 8.22 0.8.0 (2018-03-06)

Introducing Secured environment support, vastly expanded Versioning support including import/export. Fixed Templates, better documentation, more demos, and NiFi version backtesting.

### Potentially Breaking Changes

*Users should check the updated documentation and ensure their tests pass as expected*

- Import/Export of Flow Versions was reworked significantly and renamed to correct bugs and remove coding complications and be generally more obvious in its behavior
- Template upload/download reworked significantly to remove direct reliance on requests and correct bugs in some environments
- Reworked many list/get functions to be more standardised as we stabilise the approaches to certain tasks. This should not change again in future
- Standardised bad user submission on `AssertionError`, bad API submission errors on `ValueError`, and general API errors on `ApiException`. This standard should flow forwards
- Switched `ruamel.yaml` from `>15` to `<15` as advised in the project documentation, as `>15` is not considered production ready

### Known Issues

- Python2 environments with older versions of openssl may run into errors like `'SSLV3_ALERT_HANDSHAKE_FAILURE'` when working in secured environments. This is not a NiPyApi bug, it's a problem with py2/openssl which is fixed by either upgrading openssl or moving to Python3 like you know you should

### New Features

- **Added support for working with secured NiFi environments, contributed by KevDoran**
  - Added demo compatibility between `secured_connection` and console to produce a rich secured and version-controlled demo environment
  - Added many secured environment convenience functions to `security.py`
  - Integrated `tokenAuth` support throughout the low-level clients



- Added simple Docker deployment support in utils module for test, demo, and development
- Standardised all documentation on more readable docstrings and rst templates across the entire codebase
- Significantly expanded versioning support, users should consult the refreshed documentation
- Added experimental support for cleaning queues, process\_groups, and setting scheduling of various components
- Many calls now have an auto-refresh before action option to simplify applying changes
- Implemented short and long wait controls for relevant functions to allow more deterministic changes
- Implemented generic object-list-filtering-for-a-string-in-a-field for many response get/list types
- Standardised many responses to conform to a common response contract: None for none, object for single, and list-of-objects for many
- Implemented import/export to json/yaml in versioning
- Added regression/backtesting for many functions going back through major release versions to NiFi-1.1.2. More details will be obvious from reading tests/conftest.py
- Test suites now more reliably clean up after themselves when executed on long-running environments
- Apparently logging is popular, so standard Python logging is now included

#### Other notes

- Various low-level SDK bugfixes corrected in the swagger spec and updated in the provided client
- Enhanced Template and Flow Versioning to handle significantly more complex flows
- Significantly enhanced testing fixtures
- Refactored several common functions to utils.py, and moved several common configurations to config.py
- versioning.get\_flow will now export the raw Registry object for convenience when serialising flows
- Significantly improved Py2/Py3 compatibility handling, and import management within the package
- Removed docs dependency on M2R by converting everything over to reStructuredText

## 8.23 0.7.0 (2018-01-30)

- Updated project to support NiFi-1.5.0 and NiFi-Registry-0.1.0
- Merged api clients into main codebase, deprecated external client requirement
- Created centralised project configuration and test configuration
- Updated automated test environment to consistent docker for local and Travis
- Removed procedurally generated boilerplate stub tests to improve readability
- Moved pytest fixtures into conftest and expanded dramatically
- Added limited support for processor and process group scheduling
- Added support for all common Nifi-Registry calls
- Added a demo package to provide an interactive test and demo console
- Significant readme, contribution, and other documentation refresh
- Expanded CRUD support for most processor, process group and related tasks

## 8.24 0.6.1 (2018-01-04)

- Added requested functions to find and list Processors on the canvas
- Fixed list all process groups to include the root special case properly

## 8.25 0.6.0 (2017-12-31)

- Refactored many functions to use native NiFi datatypes instead of generics
- Standardised several call names for consistency
- Updated examples
- Created additional tests and enhanced existing to capture several exceptions

## 8.26 0.5.1 (2017-12-07)

- Added template import/export with working xml parsing and tests
- Added a ton of testing and validation steps
- Cleared many todos out of code by either implementing or moving to todo doc

## 8.27 0.5.0 (2017-12-06)

- migrated swagger\_client to separate repo to allow independent versions
- refactored wrapper Classes to simpler functions instead
- cleaned up documentation and project administrivia to support the split

## 8.28 0.4.0 (2017-10-29)

- Added wrapper functions for many common Template commands (templates.py)
- Added new functions for common Process Groups commands (canvas.py)
- Significant test framework enhancements for wrapper functions
- Many coding style cleanups in preparation for filling out test suite
- Added linting
- Cleaned up docs layout and placement within project
- Integrated with TravisCI
- Dropped Python2.6 testing (wasn't listed as supported anyway)
- Updated examples and Readme to be more informative

## 8.29 0.3.2 (2017-09-04)

- Fixed bug where tox failing locally due to coveralls expecting travis
- Fixed bug where TravisCI failing due to incorrectly set install requirements
- Fixed bug where swagger\_client not importing as expected

## 8.30 0.3.1 (2017-09-04)

- Fixed imports and requirements for wheel install from PyPi

## 8.31 0.3.0 (2017-09-04)

- Created basic wrapper structure for future development
- Added simple usage functions to complete todo task
- Added devnotes, updated usage, and various sundry other documentation cleanups
- Split tests into subfolders for better management and clarity
- Added Coveralls and License Badge
- Removed broken venv that ended up in project directory, added similar to ignore file
- Changed default URL in the configuration to default docker url and port on localhost

## 8.32 0.2.1 (2017-08-26)

- Fixed up removal of leftover swagger client dependencies

## 8.33 0.2.0 (2017-08-25)

- **Merge the nifi swagger client into this repo as a sub package**
  - Restructured tests into package subfolders
  - Consolidate package configuration
  - Setup package import structure
  - Updated usage instructions
  - Integrate documentation

## 8.34 0.1.2 (2017-08-24)

- Created basic integration with nifi-python-swagger-client

### 8.35 0.1.1 (2017-08-24)

- Cleaned up base project and integrations ready for code migration

### 8.36 0.1.0 (2017-08-24)

- First release on PyPI.

## CHAPTER 9

---

### Indices and tables

---

- `genindex`
- `search`



### n

nipyapi.canvas, 10  
nipyapi.config, 19  
nipyapi.nifi.api\_client, 533  
nipyapi.nifi.apis.access\_api, 36  
nipyapi.nifi.apis.connections\_api, 41  
nipyapi.nifi.apis.controller\_api, 43  
nipyapi.nifi.apis.controller\_services\_api, 51  
nipyapi.nifi.apis.counters\_api, 59  
nipyapi.nifi.apis.data\_transfer\_api, 61  
nipyapi.nifi.apis.flow\_api, 66  
nipyapi.nifi.apis.flowfile\_queues\_api, 91  
nipyapi.nifi.apis.funnel\_api, 96  
nipyapi.nifi.apis.input\_ports\_api, 98  
nipyapi.nifi.apis.labels\_api, 100  
nipyapi.nifi.apis.output\_ports\_api, 102  
nipyapi.nifi.apis.parameter\_contexts\_api, 105  
nipyapi.nifi.apis.policies\_api, 112  
nipyapi.nifi.apis.process\_groups\_api, 116  
nipyapi.nifi.apis.processors\_api, 140  
nipyapi.nifi.apis.provenance\_api, 148  
nipyapi.nifi.apis.provenance\_events\_api, 153  
nipyapi.nifi.apis.remote\_process\_groups\_api, 155  
nipyapi.nifi.apis.reporting\_tasks\_api, 162  
nipyapi.nifi.apis.resources\_api, 169  
nipyapi.nifi.apis.site\_to\_site\_api, 170  
nipyapi.nifi.apis.snippets\_api, 171  
nipyapi.nifi.apis.system\_diagnostics\_api, 173  
nipyapi.nifi.apis.templates\_api, 174  
nipyapi.nifi.apis.tenants\_api, 175  
nipyapi.nifi.apis.versions\_api, 181  
nipyapi.nifi.configuration, 535  
nipyapi.nifi.models.about\_dto, 190  
nipyapi.nifi.models.about\_entity, 192  
nipyapi.nifi.models.access\_configuration\_dto, 192  
nipyapi.nifi.models.access\_configuration\_entity, 193  
nipyapi.nifi.models.access\_policy\_dto, 194  
nipyapi.nifi.models.access\_policy\_entity, 195  
nipyapi.nifi.models.access\_policy\_summary\_dto, 197  
nipyapi.nifi.models.access\_policy\_summary\_entity, 199  
nipyapi.nifi.models.access\_status\_dto, 200  
nipyapi.nifi.models.access\_status\_entity, 201  
nipyapi.nifi.models.action\_details\_dto, 202  
nipyapi.nifi.models.action\_dto, 202  
nipyapi.nifi.models.action\_entity, 204  
nipyapi.nifi.models.activate\_controller\_services\_entity, 205  
nipyapi.nifi.models.affected\_component\_dto, 206  
nipyapi.nifi.models.affected\_component\_entity, 207  
nipyapi.nifi.models.allowable\_value\_dto, 209  
nipyapi.nifi.models.allowable\_value\_entity, 210  
nipyapi.nifi.models.attribute\_dto, 210  
nipyapi.nifi.models.banner\_dto, 211  
nipyapi.nifi.models.banner\_entity, 212  
nipyapi.nifi.models.batch\_settings\_dto, 212  
nipyapi.nifi.models.batch\_size, 213  
nipyapi.nifi.models.bucket, 214

nipyapi.nifi.models.bucket\_dto, 216  
nipyapi.nifi.models.bucket\_entity, 216  
nipyapi.nifi.models.buckets\_entity, 217  
nipyapi.nifi.models.bulletin\_board\_dto, 218  
nipyapi.nifi.models.bulletin\_board\_entity, 218  
nipyapi.nifi.models.bulletin\_dto, 219  
nipyapi.nifi.models.bulletin\_entity, 221  
nipyapi.nifi.models.bundle, 222  
nipyapi.nifi.models.bundle\_dto, 223  
nipyapi.nifi.models.cluste\_summary\_entity, 223  
nipyapi.nifi.models.cluster\_dto, 224  
nipyapi.nifi.models.cluster\_entity, 225  
nipyapi.nifi.models.cluster\_search\_results\_entity, 225  
nipyapi.nifi.models.cluster\_summary\_dto, 226  
nipyapi.nifi.models.component\_details\_dto, 227  
nipyapi.nifi.models.component\_difference\_dto, 227  
nipyapi.nifi.models.component\_history\_dto, 229  
nipyapi.nifi.models.component\_history\_entity, 229  
nipyapi.nifi.models.component\_reference\_dto, 230  
nipyapi.nifi.models.component\_reference\_entity, 231  
nipyapi.nifi.models.component\_search\_result\_dto, 233  
nipyapi.nifi.models.component\_state\_dto, 234  
nipyapi.nifi.models.component\_state\_entity, 235  
nipyapi.nifi.models.connectable\_component, 236  
nipyapi.nifi.models.connectable\_dto, 237  
nipyapi.nifi.models.connection\_dto, 238  
nipyapi.nifi.models.connection\_entity, 242  
nipyapi.nifi.models.connection\_status\_dto, 244  
nipyapi.nifi.models.connection\_status\_entity, 246  
nipyapi.nifi.models.connection\_status\_snapshot\_dto, 247  
nipyapi.nifi.models.connection\_status\_snapshot\_entity, 251  
nipyapi.nifi.models.connections\_entity, 252  
nipyapi.nifi.models.controller\_bulletins\_entity, 252  
nipyapi.nifi.models.controller\_configuration\_dto, 253  
nipyapi.nifi.models.controller\_configuration\_entity, 254  
nipyapi.nifi.models.controller\_dto, 255  
nipyapi.nifi.models.controller\_entity, 258  
nipyapi.nifi.models.controller\_service\_api, 259  
nipyapi.nifi.models.controller\_service\_api\_dto, 259  
nipyapi.nifi.models.controller\_service\_dto, 260  
nipyapi.nifi.models.controller\_service\_entity, 265  
nipyapi.nifi.models.controller\_service\_referencing\_entity, 267  
nipyapi.nifi.models.controller\_service\_referencing\_entity, 269  
nipyapi.nifi.models.controller\_service\_referencing\_entity, 271  
nipyapi.nifi.models.controller\_service\_types\_entity, 272  
nipyapi.nifi.models.controller\_services\_entity, 272  
nipyapi.nifi.models.controller\_status\_dto, 273  
nipyapi.nifi.models.controller\_status\_entity, 276  
nipyapi.nifi.models.copy\_snippet\_request\_entity, 277  
nipyapi.nifi.models.counter\_dto, 278  
nipyapi.nifi.models.counter\_entity, 279  
nipyapi.nifi.models.counters\_dto, 279  
nipyapi.nifi.models.counters\_entity, 280  
nipyapi.nifi.models.counters\_snapshot\_dto, 281  
nipyapi.nifi.models.create\_active\_request\_entity, 281  
nipyapi.nifi.models.create\_template\_request\_entity, 282  
nipyapi.nifi.models.current\_user\_entity, 283  
nipyapi.nifi.models.difference\_dto, 285  
nipyapi.nifi.models.dimensions\_dto, 286  
nipyapi.nifi.models.documented\_type\_dto, 287  
nipyapi.nifi.models.drop\_request\_dto, 288  
nipyapi.nifi.models.drop\_request\_entity, 291  
nipyapi.nifi.models.flow\_breadcrumb\_dto, 292



---

nipyapi.nifi.models.flow_breadcrumb_entity,	nipyapi.nifi.models.node_remote_process_group_status_dto,
293	331
nipyapi.nifi.models.flow_comparison_entity,	nipyapi.nifi.models.node_search_result_dto,
294	332
nipyapi.nifi.models.flow_configuration_dto,	nipyapi.nifi.models.node_status_snapshots_dto,
294	332
nipyapi.nifi.models.flow_configuration_entity,	nipyapi.nifi.models.node_system_diagnostics_snapshot_dto,
296	333
nipyapi.nifi.models.flow_dto,	nipyapi.nifi.models.output_ports_entity,
297	334
nipyapi.nifi.models.flow_entity,	nipyapi.nifi.models.peer_dto,
298	335
nipyapi.nifi.models.flow_file_dto,	nipyapi.nifi.models.peers_entity,
298	336
nipyapi.nifi.models.flow_file_entity,	nipyapi.nifi.models.permissions,
301	336
nipyapi.nifi.models.flow_file_summary_dto,	nipyapi.nifi.models.permissions_dto,
302	337
nipyapi.nifi.models.flow_snippet_dto,	nipyapi.nifi.models.port_dto,
304	338
nipyapi.nifi.models.funnel_dto,	nipyapi.nifi.models.port_entity,
305	340
nipyapi.nifi.models.funnel_entity,	nipyapi.nifi.models.port_status_dto,
306	342
nipyapi.nifi.models.funnels_entity,	nipyapi.nifi.models.port_status_entity,
307	343
nipyapi.nifi.models.garbage_collection_dto,	nipyapi.nifi.models.port_status_snapshot_dto,
308	344
nipyapi.nifi.models.history_dto,	nipyapi.nifi.models.port_status_snapshot_entity,
309	346
nipyapi.nifi.models.history_entity,	nipyapi.nifi.models.position_dto,
310	347
nipyapi.nifi.models.input_ports_entity,	nipyapi.nifi.models.previous_value_dto,
310	347
nipyapi.nifi.models.instantiate_template_entity,	nipyapi.nifi.models.prioritizer_types_entity,
311	348
nipyapi.nifi.models.label_dto,	nipyapi.nifi.models.process_group_dto,
312	349
nipyapi.nifi.models.label_entity,	nipyapi.nifi.models.process_group_entity,
314	355
nipyapi.nifi.models.labels_entity,	nipyapi.nifi.models.process_group_flow_dto,
315	360
nipyapi.nifi.models.lineage_dto,	nipyapi.nifi.models.process_group_flow_entity,
316	362
nipyapi.nifi.models.lineage_entity,	nipyapi.nifi.models.process_group_status_dto,
317	362
nipyapi.nifi.models.lineage_request_dto,	nipyapi.nifi.models.process_group_status_entity,
318	363
nipyapi.nifi.models.lineage_results_dto,	nipyapi.nifi.models.process_group_status_snapshot_dto,
319	364
nipyapi.nifi.models.listing_request_dto,	nipyapi.nifi.models.process_group_status_snapshot_entity,
320	370
nipyapi.nifi.models.listing_request_entity,	nipyapi.nifi.models.process_groups_entity,
322	371
nipyapi.nifi.models.node_connection_status_snapshot_dto,	nipyapi.nifi.models.processor_config_dto,
323	371
nipyapi.nifi.models.node_counters_snapshot_dto,	nipyapi.nifi.models.processor_dto,
324	375
nipyapi.nifi.models.node_dto,	nipyapi.nifi.models.processor_entity,
325	375
nipyapi.nifi.models.node_entity,	nipyapi.nifi.models.processor_status_dto,
326	381
nipyapi.nifi.models.node_event_dto,	nipyapi.nifi.models.processor_status_entity,
327	
nipyapi.nifi.models.node_port_status_snapshot_dto,	
328	
nipyapi.nifi.models.node_process_group_status_snapshot_dto,	
329	
nipyapi.nifi.models.node_processor_status_snapshot_dto,	
330	

382 nipyapi.nifi.models.remote\_process\_group\_status\_dto,  
nipyapi.nifi.models.processor\_status\_snapshot\_dto, 428  
383 nipyapi.nifi.models.remote\_process\_group\_status\_entity,  
nipyapi.nifi.models.processor\_status\_snapshot\_entity, 430  
387 nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto,  
nipyapi.nifi.models.processor\_types\_entity, 430  
388 nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity,  
nipyapi.nifi.models.processors\_entity, 433  
388 nipyapi.nifi.models.remote\_process\_groups\_entity,  
nipyapi.nifi.models.property\_descriptor\_dto, 433  
389 nipyapi.nifi.models.reporting\_task\_dto,  
nipyapi.nifi.models.property\_descriptor\_entity, 434  
392 nipyapi.nifi.models.reporting\_task\_entity,  
nipyapi.nifi.models.property\_history\_dto, 438  
392 nipyapi.nifi.models.reporting\_task\_types\_entity,  
nipyapi.nifi.models.provenance\_dto, 393 440  
nipyapi.nifi.models.provenance\_entity, nipyapi.nifi.models.reporting\_tasks\_entity,  
394 441  
nipyapi.nifi.models.provenance\_event\_dto, nipyapi.nifi.models.resource\_dto, 441  
395 nipyapi.nifi.models.resources\_entity,  
nipyapi.nifi.models.provenance\_event\_entity, 442  
402 nipyapi.nifi.models.revision\_dto, 443  
nipyapi.nifi.models.provenance\_link\_dto, nipyapi.nifi.models.schedule\_components\_entity,  
402 443  
nipyapi.nifi.models.provenance\_node\_dto, nipyapi.nifi.models.search\_results\_dto,  
403 444  
nipyapi.nifi.models.provenance\_options\_dto, nipyapi.nifi.models.search\_results\_entity,  
405 446  
nipyapi.nifi.models.provenance\_options\_entity, nipyapi.nifi.models.snippet\_dto, 447  
406 nipyapi.nifi.models.snippet\_entity, 449  
nipyapi.nifi.models.provenance\_request\_dto, nipyapi.nifi.models.start\_version\_control\_request\_dto,  
406 450  
nipyapi.nifi.models.provenance\_results\_dto, nipyapi.nifi.models.state\_entry\_dto, 451  
408 nipyapi.nifi.models.state\_map\_dto, 451  
nipyapi.nifi.models.provenance\_searchable\_entity\_dto, nipyapi.nifi.models.status\_descriptor\_dto,  
410 452  
nipyapi.nifi.models.queue\_size\_dto, 411 nipyapi.nifi.models.status\_history\_dto,  
nipyapi.nifi.models.registry\_client\_entity, 453  
411 nipyapi.nifi.models.status\_history\_entity,  
nipyapi.nifi.models.registry\_clients\_entity, 455  
413 nipyapi.nifi.models.status\_snapshot\_dto,  
nipyapi.nifi.models.registry\_dto, 414 455  
nipyapi.nifi.models.relationship\_dto, nipyapi.nifi.models.storage\_usage\_dto,  
414 456  
nipyapi.nifi.models.remote\_process\_group\_components\_dto, nipyapi.nifi.models.streaming\_output,  
415 457  
nipyapi.nifi.models.remote\_process\_group\_components\_entity, nipyapi.nifi.models.submit\_replay\_request\_entity,  
416 458  
nipyapi.nifi.models.remote\_process\_group\_components\_entity, nipyapi.nifi.models.system\_diagnostics\_dto,  
421 459  
nipyapi.nifi.models.remote\_process\_group\_components\_entity, nipyapi.nifi.models.system\_diagnostics\_entity,  
424 459  
nipyapi.nifi.models.remote\_process\_group\_components\_entity, nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto,  
426 460

[nipyapi.nifi.models.template\\_dto, 465](#)  
[nipyapi.nifi.models.template\\_entity, 466](#)  
[nipyapi.nifi.models.templates\\_entity, 468](#)  
[nipyapi.nifi.models.tenant\\_dto, 468](#)  
[nipyapi.nifi.models.tenant\\_entity, 469](#)  
[nipyapi.nifi.models.tenants\\_entity, 471](#)  
[nipyapi.nifi.models.transaction\\_result\\_entity, 471](#)  
[nipyapi.nifi.models.update\\_controller\\_service\\_reference\\_request\\_entity, 472](#)  
[nipyapi.nifi.models.user\\_dto, 474](#)  
[nipyapi.nifi.models.user\\_entity, 475](#)  
[nipyapi.nifi.models.user\\_group\\_dto, 476](#)  
[nipyapi.nifi.models.user\\_group\\_entity, 478](#)  
[nipyapi.nifi.models.user\\_groups\\_entity, 479](#)  
[nipyapi.nifi.models.users\\_entity, 480](#)  
[nipyapi.nifi.models.variable\\_dto, 481](#)  
[nipyapi.nifi.models.variable\\_entity, 481](#)  
[nipyapi.nifi.models.variable\\_registry\\_dto, 482](#)  
[nipyapi.nifi.models.variable\\_registry\\_entity, 483](#)  
[nipyapi.nifi.models.variable\\_registry\\_update\\_controller\\_service\\_reference\\_request\\_entity, 484](#)  
[nipyapi.nifi.models.variable\\_registry\\_update\\_controller\\_service\\_reference\\_request\\_entity, 486](#)  
[nipyapi.nifi.models.variable\\_registry\\_update\\_status\\_dto, 487](#)  
[nipyapi.nifi.models.version\\_control\\_component\\_entity, 487](#)  
[nipyapi.nifi.models.version\\_control\\_info\\_entity, 489](#)  
[nipyapi.nifi.models.version\\_control\\_info\\_entity, 491](#)  
[nipyapi.nifi.models.version\\_info\\_dto, 491](#)  
[nipyapi.nifi.models.versioned\\_connection, 493](#)  
[nipyapi.nifi.models.versioned\\_controller\\_service\\_reference\\_request\\_entity, 497](#)  
[nipyapi.nifi.models.versioned\\_flow, 499](#)  
[nipyapi.nifi.models.versioned\\_flow\\_coordination\\_request\\_entity, 501](#)  
[nipyapi.nifi.models.versioned\\_flow\\_dto, 503](#)  
[nipyapi.nifi.models.versioned\\_flow\\_entity, 504](#)  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_entity, 504](#)  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_entity, 506](#)  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_metadata\\_entity, 507](#)  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_metadata\\_entity, 508](#)  
[nipyapi.nifi.models.versioned\\_flow\\_update\\_request\\_entity, 509](#)  
[nipyapi.nifi.models.versioned\\_flow\\_update\\_request\\_entity, 510](#)  
[nipyapi.nifi.models.versioned\\_flows\\_entity, 511](#)  
[nipyapi.nifi.models.versioned\\_funnel, 512](#)  
[nipyapi.nifi.models.versioned\\_label, 513](#)  
[nipyapi.nifi.models.versioned\\_port, 515](#)  
[nipyapi.nifi.models.versioned\\_process\\_group, 517](#)  
[nipyapi.nifi.models.versioned\\_processor, 521](#)  
[nipyapi.nifi.models.versioned\\_property\\_descriptor, 526](#)  
[nipyapi.nifi.models.versioned\\_remote\\_group\\_port, 527](#)  
[nipyapi.nifi.models.versioned\\_remote\\_process\\_group, 529](#)  
[nipyapi.nifi.rest, 536](#)  
[nipyapi.registry.api\\_client, 619](#)  
[nipyapi.registry.apis.access\\_api, 537](#)  
[nipyapi.registry.apis.bucket\\_flows\\_api, 543](#)  
[nipyapi.registry.apis.buckets\\_api, 551](#)  
[nipyapi.registry.apis.flows\\_api, 554](#)  
[nipyapi.registry.apis.items\\_api, 557](#)  
[nipyapi.registry.apis.policies\\_api, 559](#)  
[nipyapi.registry.apis.tenants\\_api, 562](#)  
[nipyapi.registry.configuration, 621](#)  
[nipyapi.registry.models.access\\_policy, 568](#)  
[nipyapi.registry.models.access\\_policy\\_summary, 569](#)  
[nipyapi.registry.models.batch\\_size, 571](#)  
[nipyapi.registry.models.bucket, 571](#)  
[nipyapi.registry.models.bucket\\_item, 573](#)  
[nipyapi.registry.models.bundle, 575](#)  
[nipyapi.registry.models.connectable\\_component, 575](#)  
[nipyapi.registry.models.controller\\_service\\_api, 577](#)  
[nipyapi.registry.models.current\\_user, 577](#)  
[nipyapi.registry.models.fields, 578](#)  
[nipyapi.registry.models.permissions, 579](#)  
[nipyapi.registry.models.resource, 580](#)  
[nipyapi.registry.models.resource\\_permissions, 580](#)

580  
nipyapi.registry.models.tenant, 581  
nipyapi.registry.models.user, 583  
nipyapi.registry.models.user\_group, 584  
nipyapi.registry.models.versioned\_connection,  
585  
nipyapi.registry.models.versioned\_controller\_service,  
589  
nipyapi.registry.models.versioned\_flow,  
592  
nipyapi.registry.models.versioned\_flow\_coordinates,  
594  
nipyapi.registry.models.versioned\_flow\_snapshot,  
595  
nipyapi.registry.models.versioned\_flow\_snapshot\_metadata,  
597  
nipyapi.registry.models.versioned\_funnel,  
598  
nipyapi.registry.models.versioned\_label,  
599  
nipyapi.registry.models.versioned\_port,  
601  
nipyapi.registry.models.versioned\_process\_group,  
603  
nipyapi.registry.models.versioned\_processor,  
607  
nipyapi.registry.models.versioned\_property\_descriptor,  
612  
nipyapi.registry.models.versioned\_remote\_group\_port,  
613  
nipyapi.registry.models.versioned\_remote\_process\_group,  
615  
nipyapi.registry.rest, 621  
nipyapi.security, 21  
nipyapi.system, 25  
nipyapi.templates, 26  
nipyapi.utils, 28  
nipyapi.versioning, 31

## A

- about (*nipyapi.nifi.models.about\_entity.AboutEntity* attribute), 192
- AboutDTO (class in *nipyapi.nifi.models.about\_dto*), 191
- AboutEntity (class in *nipyapi.nifi.models.about\_entity*), 192
- access\_policies (*nipyapi.nifi.models.user\_dto.UserDTO* attribute), 474
- access\_policies (*nipyapi.nifi.models.user\_group\_dto.UserGroupDTO* attribute), 477
- access\_policies (*nipyapi.registry.models.tenant.Tenant* attribute), 582
- access\_policies (*nipyapi.registry.models.user.User* attribute), 583
- access\_policies (*nipyapi.registry.models.user\_group.UserGroup* attribute), 584
- access\_status (*nipyapi.nifi.models.access\_status\_entity.AccessStatusEntity* attribute), 201
- AccessApi (class in *nipyapi.nifi.apis.access\_api*), 36
- AccessApi (class in *nipyapi.registry.apis.access\_api*), 537
- AccessConfigurationDTO (class in *nipyapi.nifi.models.access\_configuration\_dto*), 193
- AccessConfigurationEntity (class in *nipyapi.nifi.models.access\_configuration\_entity*), 193
- AccessPolicy (class in *nipyapi.registry.models.access\_policy*), 568
- AccessPolicyDTO (class in *nipyapi.nifi.models.access\_policy\_dto*), 194
- AccessPolicyEntity (class in *nipyapi.nifi.models.access\_policy\_entity*), 195
- AccessPolicySummary (class in *nipyapi.registry.models.access\_policy\_summary*), 569
- AccessPolicySummaryDTO (class in *nipyapi.nifi.models.access\_policy\_summary\_dto*), 197
- AccessPolicySummaryEntity (class in *nipyapi.nifi.models.access\_policy\_summary\_entity*), 199
- AccessStatusDTO (class in *nipyapi.nifi.models.access\_status\_dto*), 200
- AccessStatusEntity (class in *nipyapi.nifi.models.access\_status\_entity*), 201
- action (*nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO* attribute), 194
- action (*nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO* attribute), 198
- action (*nipyapi.nifi.models.action\_entity.ActionEntity* attribute), 204
- action (*nipyapi.nifi.models.versioned\_flow\_dto.VersionedFlowDTO* attribute), 503
- action (*nipyapi.registry.models.access\_policy.AccessPolicy* attribute), 568
- action (*nipyapi.registry.models.access\_policy\_summary.AccessPolicySummary* attribute), 570
- action\_details (*nipyapi.nifi.models.action\_dto.ActionDTO* attribute), 202
- ActionDetailsDTO (class in *nipyapi.nifi.models.action\_details\_dto*), 202
- ActionDTO (class in *nipyapi.nifi.models.action\_dto*), 202
- ActionEntity (class in *nipyapi.nifi.models.action\_entity*), 204
- actions (*nipyapi.nifi.models.history\_dto.HistoryDTO* attribute), 309
- activate\_controller\_services () (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 66
- activate\_controller\_services\_with\_http\_info () (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 66
- ActivateControllerServicesEntity (class in *nipyapi.nifi.models.activate\_controller\_services\_entity*), 205

active\_remote\_input\_port\_count (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 418  
 active\_remote\_output\_port\_count (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 418  
 active\_remote\_port\_count (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 256  
 active\_remote\_port\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO attribute), 274  
 active\_remote\_port\_count (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 351  
 active\_remote\_port\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 357  
 active\_thread\_count (nipyapi.nifi.models.affected\_component\_dto.AffectedComponentDTO attribute), 206  
 active\_thread\_count (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO attribute), 267  
 active\_thread\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO attribute), 274  
 active\_thread\_count (nipyapi.nifi.models.node\_dto.NodeDTO attribute), 325  
 active\_thread\_count (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO attribute), 344  
 active\_thread\_count (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 366  
 active\_thread\_count (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 384  
 active\_thread\_count (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshotDTO attribute), 431  
 active\_thread\_count (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 435  
 add\_user\_group\_to\_access\_policy() (in module nipyapi.security), 24  
 add\_user\_to\_access\_policy() (in module nipyapi.security), 22  
 address (nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto.NodeConnectionStatusSnapshotDTO attribute), 323  
 address (nipyapi.nifi.models.node\_counters\_snapshot\_dto.NodeCountersSnapshotDTO attribute), 324  
 address (nipyapi.nifi.models.node\_dto.NodeDTO attribute), 214  
 address (nipyapi.nifi.models.node\_port\_status\_snapshot\_dto.NodePortStatusSnapshotDTO attribute), 328  
 address (nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.NodeProcessorStatusSnapshotDTO attribute), 330  
 address (nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto.NodeRemoteProcessGroupStatusSnapshotDTO attribute), 331  
 address (nipyapi.nifi.models.node\_search\_result\_dto.NodeSearchResultDTO attribute), 332  
 address (nipyapi.nifi.models.node\_status\_snapshots\_dto.NodeStatusSnapshotsDTO attribute), 333  
 address (nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto.NodeSystemDiagnosticsSnapshotDTO attribute), 334  
 affected\_components (nipyapi.nifi.models.variable\_dto.VariableDTO attribute), 481  
 affected\_components (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO attribute), 484  
 AffectedComponentDTO (class in nipyapi.nifi.models.affected\_component\_entity), 206  
 AffectedComponentEntity (class in nipyapi.nifi.models.affected\_component\_entity), 207  
 aggregate\_snapshot (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO attribute), 245  
 aggregate\_snapshot (nipyapi.nifi.models.counters\_dto.CountersDTO attribute), 279  
 aggregate\_snapshot (nipyapi.nifi.models.port\_status\_dto.PortStatusDTO attribute), 342  
 aggregate\_snapshot (nipyapi.nifi.models.process\_group\_status\_dto.ProcessGroupStatusDTO attribute), 363  
 aggregate\_snapshot (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 381  
 aggregate\_snapshot (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO attribute), 428  
 aggregate\_snapshot (nipyapi.nifi.models.system\_diagnostics\_dto.SystemDiagnosticsDTO attribute), 459  
 aggregate\_snapshots (nipyapi.nifi.models.status\_snapshot\_dto.StatusHistoryDTO attribute), 454  
 NodeCountersSnapshotDTO (nipyapi.nifi.models.bucket.Bucket attribute), 214



allow\_bundle\_redeploy  
(*nipyapi.registry.models.bucket.Bucket* attribute), 572

allow\_public\_read  
(*nipyapi.nifi.models.bucket.Bucket* attribute), 214

allow\_public\_read  
(*nipyapi.registry.models.bucket.Bucket* attribute), 572

allow\_remote\_access  
(*nipyapi.nifi.models.port\_dto.PortDTO* attribute), 338

allow\_remote\_access  
(*nipyapi.nifi.models.port\_entity.PortEntity* attribute), 340

allow\_remote\_access  
(*nipyapi.nifi.models.versioned\_port.VersionedPort* attribute), 515

allow\_remote\_access  
(*nipyapi.registry.models.versioned\_port.VersionedPort* attribute), 602

allowable\_value (*nipyapi.nifi.models.allowable\_value\_entity.AllowableValueEntity* attribute), 210

allowable\_values (*nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDTO* attribute), 390

AllowableValueDTO (class in *nipyapi.nifi.models.allowable\_value\_dto*), 209

AllowableValueEntity (class in *nipyapi.nifi.models.allowable\_value\_entity*), 210

alternate\_identifier\_uri  
(*nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO* attribute), 397

analyze\_configuration()  
(*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi* method), 51

analyze\_configuration()  
(*nipyapi.nifi.apis.processors\_api.ProcessorsApi* method), 140

analyze\_configuration()  
(*nipyapi.nifi.apis.reporting\_tasks\_api.ReportingTasksApi* method), 162

analyze\_configuration\_with\_http\_info()  
(*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi* method), 51

analyze\_configuration\_with\_http\_info()  
(*nipyapi.nifi.apis.processors\_api.ProcessorsApi* method), 140

analyze\_configuration\_with\_http\_info()  
(*nipyapi.nifi.apis.reporting\_tasks\_api.ReportingTasksApi* method), 162

annotation\_data (*nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO* attribute), 262

annotation\_data (*nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO* attribute), 372

annotation\_data (*nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO* attribute), 435

annotation\_data (*nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService* attribute), 498

annotation\_data (*nipyapi.nifi.models.versioned\_processor.VersionedProcessor* attribute), 522

annotation\_data (*nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService* attribute), 590

annotation\_data (*nipyapi.registry.models.versioned\_processor.VersionedProcessor* attribute), 608

anonymous (*nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity* attribute), 284

anonymous (*nipyapi.registry.models.current\_user.CurrentUser* attribute), 578

any\_top\_level\_resource  
(*nipyapi.registry.models.resource\_permissions.ResourcePermissions* attribute), 581

api\_port (*nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto.NodeConnectionStatusSnapshotDTO* attribute), 323

api\_port (*nipyapi.nifi.models.node\_counters\_snapshot\_dto.NodeCountersSnapshotDTO* attribute), 324

api\_port (*nipyapi.nifi.models.node\_dto.NodeDTO* attribute), 325

api\_port (*nipyapi.nifi.models.node\_port\_status\_snapshot\_dto.NodePortStatusSnapshotDTO* attribute), 328

api\_port (*nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto.NodeProcessGroupStatusSnapshotDTO* attribute), 329

api\_port (*nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.NodeProcessorStatusSnapshotDTO* attribute), 330

api\_port (*nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto.NodeRemoteProcessGroupStatusSnapshotDTO* attribute), 331

api\_port (*nipyapi.nifi.models.node\_status\_snapshots\_dto.NodeStatusSnapshotsDTO* attribute), 333

api\_port (*nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto.NodeSystemDiagnosticsSnapshotDTO* attribute), 334

ApiClient (class in *nipyapi.nifi.api\_client*), 533

ApiClient (class in *nipyapi.registry.api\_client*), 619

ApiException, 536, 622

artifact (*nipyapi.nifi.models.bundle.Bundle* attribute), 222

artifact (*nipyapi.nifi.models.bundle\_dto.BundleDTO* attribute), 223

artifact (*nipyapi.registry.models.bundle.Bundle* attribute), 575

assign\_context\_to\_process\_group() (in module *nipyapi.parameters*), 21

attribute\_map (*nipyapi.nifi.models.about\_dto>AboutDTO* attribute), 191

attribute\_map (*nipyapi.nifi.models.about\_entity>AboutEntity* attribute), 192

attribute\_map (*nipyapi.nifi.models.access\_configuration\_dto.AccessConfigurationDTO* attribute), 193

`attribute_map(nipyapi.nifi.models.access_configuration_entity.AccessConfigurationEntity, attribute), 193`  
`attribute_map(nipyapi.nifi.models.access_policy_dto.AccessPolicyDTO, attribute), 194`  
`attribute_map(nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity, attribute), 196`  
`attribute_map(nipyapi.nifi.models.access_policy_summary_dto.AccessPolicySummaryDTO, attribute), 198`  
`attribute_map(nipyapi.nifi.models.access_policy_summary_entity.AccessPolicySummaryEntity, attribute), 199`  
`attribute_map(nipyapi.nifi.models.access_status_dto.AccessStatusDTO, attribute), 201`  
`attribute_map(nipyapi.nifi.models.access_status_entity.AccessStatusEntity, attribute), 201`  
`attribute_map(nipyapi.nifi.models.action_details_dto.ActionDetailsDTO, attribute), 202`  
`attribute_map(nipyapi.nifi.models.action_dto.ActionDTO, attribute), 203`  
`attribute_map(nipyapi.nifi.models.action_entity.ActionEntity, attribute), 204`  
`attribute_map(nipyapi.nifi.models.activate_controller_services_entity.ActivateControllerServicesEntity, attribute), 205`  
`attribute_map(nipyapi.nifi.models.affected_component_dto.AffectedComponentDTO, attribute), 206`  
`attribute_map(nipyapi.nifi.models.affected_component_entity.AffectedComponentEntity, attribute), 208`  
`attribute_map(nipyapi.nifi.models.allowable_value_dto.AllowableValueDTO, attribute), 209`  
`attribute_map(nipyapi.nifi.models.allowable_value_entity.AllowableValueEntity, attribute), 210`  
`attribute_map(nipyapi.nifi.models.attribute_dto.AttributeDTO, attribute), 211`  
`attribute_map(nipyapi.nifi.models.banner_dto.BannerDTO, attribute), 211`  
`attribute_map(nipyapi.nifi.models.banner_entity.BannerEntity, attribute), 212`  
`attribute_map(nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO, attribute), 213`  
`attribute_map(nipyapi.nifi.models.batch_size.BatchSize, attribute), 214`  
`attribute_map(nipyapi.nifi.models.bucket.Bucket, attribute), 215`  
`attribute_map(nipyapi.nifi.models.bucket_dto.BucketDTO, attribute), 216`  
`attribute_map(nipyapi.nifi.models.bucket_entity.BucketEntity, attribute), 217`  
`attribute_map(nipyapi.nifi.models.buckets_entity.BucketsEntity, attribute), 217`  
`attribute_map(nipyapi.nifi.models.bulletin_board_dto.BulletinBoardDTO, attribute), 218`  
`attribute_map(nipyapi.nifi.models.bulletin_board_entity.BulletinBoardEntity, attribute), 219`  
`attribute_map(nipyapi.nifi.models.bulletin_dto.BulletinDTO, attribute), 219`  
`attribute_map(nipyapi.nifi.models.bulletin_entity.BulletinEntity, attribute), 221`  
`attribute_map(nipyapi.nifi.models.bundle.Bundle, attribute), 222`  
`attribute_map(nipyapi.nifi.models.bundle_dto.BundleDTO, attribute), 223`  
`attribute_map(nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO, attribute), 224`  
`attribute_map(nipyapi.nifi.models.cluster_summary_entity.ClusterSummaryEntity, attribute), 224`  
`attribute_map(nipyapi.nifi.models.cluster_dto.ClusterDTO, attribute), 224`  
`attribute_map(nipyapi.nifi.models.cluster_entity.ClusterEntity, attribute), 225`  
`attribute_map(nipyapi.nifi.models.cluster_search_results_entity.ClusterSearchResultsEntity, attribute), 225`  
`attribute_map(nipyapi.nifi.models.component_details_dto.ComponentDetailsDTO, attribute), 227`  
`attribute_map(nipyapi.nifi.models.component_difference_dto.ComponentDifferenceDTO, attribute), 228`  
`attribute_map(nipyapi.nifi.models.component_history_dto.ComponentHistoryDTO, attribute), 229`  
`attribute_map(nipyapi.nifi.models.component_history_entity.ComponentHistoryEntity, attribute), 229`  
`attribute_map(nipyapi.nifi.models.component_reference_dto.ComponentReferenceDTO, attribute), 230`  
`attribute_map(nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity, attribute), 231`  
`attribute_map(nipyapi.nifi.models.component_search_result_dto.ComponentSearchResultDTO, attribute), 233`  
`attribute_map(nipyapi.nifi.models.component_state_dto.ComponentStateDTO, attribute), 234`  
`attribute_map(nipyapi.nifi.models.component_state_entity.ComponentStateEntity, attribute), 235`  
`attribute_map(nipyapi.nifi.models.connectable_component.ConnectableComponent, attribute), 236`  
`attribute_map(nipyapi.nifi.models.connectable_dto.ConnectableDTO, attribute), 237`  
`attribute_map(nipyapi.nifi.models.connection_dto.ConnectionDTO, attribute), 239`  
`attribute_map(nipyapi.nifi.models.connection_entity.ConnectionEntity, attribute), 242`  
`attribute_map(nipyapi.nifi.models.connection_status_dto.ConnectionStatusDTO, attribute), 245`  
`attribute_map(nipyapi.nifi.models.connection_status_entity.ConnectionStatusEntity, attribute), 246`  
`attribute_map(nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO, attribute), 248`  
`attribute_map(nipyapi.nifi.models.connection_status_snapshot_entity.ConnectionStatusSnapshotEntity, attribute), 251`  
`attribute_map(nipyapi.nifi.models.connections_entity.ConnectionsEntity, attribute), 252`  
`attribute_map(nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity, attribute), 253`



[attribute\\_map \(nipyapi.nifi.models.controller\\_configuration\\_dto.ControllerConfigurationDTO.drop\\_request\\_dto.DropRequestDTO attribute\), 254](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_configuration\\_entity.ControllerConfigurationEntity.drop\\_request\\_entity.DropRequestEntity attribute\), 254](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_dto.ControllerDTO.to\\_map \(nipyapi.nifi.models.flow\\_breadcrumb\\_dto.FlowBreadcrumbDTO attribute\), 256](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_dto.ControllerDTO.to\\_map \(nipyapi.nifi.models.flow\\_breadcrumb\\_entity.FlowBreadcrumbEntity attribute\), 258](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_api\\_controller\\_service\\_api\\_dto.ControllerServiceApiDTO.models.flow\\_comparison\\_entity.FlowComparisonEntity attribute\), 259](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_api\\_dto.ControllerServiceApiDTO.models.flow\\_configuration\\_dto.FlowConfigurationDTO attribute\), 259](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_dto.ControllerServiceDTO.models.flow\\_configuration\\_entity.FlowConfigurationEntity attribute\), 262](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_entity.ControllerServiceEntity.models.flow\\_dto.FlowDTO attribute\), 265](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_entity.ControllerServiceReferencingComponentEntity.models.service\\_referencing\\_flow\\_component\\_dto.ServiceReferencingFlowComponentDTO attribute\), 267](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_entity.ControllerServiceReferencingComponentEntity.models.service\\_referencing\\_flow\\_component\\_entity.ServiceReferencingFlowComponentEntity attribute\), 270](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_entity.ControllerServiceReferencingComponentEntity.models.service\\_referencing\\_flow\\_entity.ServiceReferencingFlowEntity attribute\), 271](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_service\\_types\\_entity.ControllerServiceTypesEntity.models.flow\\_file\\_summary\\_dto.FlowFileSummaryDTO attribute\), 272](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_services\\_entity.ControllerServicesEntity.models.flow\\_snippet\\_dto.FlowSnippetDTO attribute\), 273](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_status\\_dto.ControllerStatusDTO.models.funnel\\_dto.FunnelDTO attribute\), 274](#)  
[attribute\\_map \(nipyapi.nifi.models.controller\\_status\\_entity.ControllerStatusEntity.models.funnel\\_entity.FunnelEntity attribute\), 276](#)  
[attribute\\_map \(nipyapi.nifi.models.copy\\_snippet\\_request\\_entity.CopySnippetRequestEntity.models.funnels\\_entity.FunnelsEntity attribute\), 277](#)  
[attribute\\_map \(nipyapi.nifi.models.counter\\_dto.CounterDTO.attribute\\_map \(nipyapi.nifi.models.garbage\\_collection\\_dto.GarbageCollectionDTO attribute\), 278](#)  
[attribute\\_map \(nipyapi.nifi.models.counter\\_entity.CounterEntity.attribute\\_map \(nipyapi.nifi.models.history\\_dto.HistoryDTO attribute\), 279](#)  
[attribute\\_map \(nipyapi.nifi.models.counters\\_dto.CountersDTO.attribute\\_map \(nipyapi.nifi.models.history\\_entity.HistoryEntity attribute\), 280](#)  
[attribute\\_map \(nipyapi.nifi.models.counters\\_entity.CountersEntity.attribute\\_map \(nipyapi.nifi.models.input\\_ports\\_entity.InputPortsEntity attribute\), 280](#)  
[attribute\\_map \(nipyapi.nifi.models.counters\\_snapshot\\_dto.CountersSnapshotDTO.attribute\\_map \(nipyapi.nifi.models.instantiate\\_template\\_request\\_entity.InstantiateTemplateRequestEntity attribute\), 281](#)  
[attribute\\_map \(nipyapi.nifi.models.create\\_active\\_request\\_entity.CreateActiveRequestEntity.models.label\\_dto.LabelDTO attribute\), 282](#)  
[attribute\\_map \(nipyapi.nifi.models.create\\_template\\_request\\_entity.CreateTemplateRequestEntity.models.label\\_entity.LabelEntity attribute\), 282](#)  
[attribute\\_map \(nipyapi.nifi.models.current\\_user\\_entity.CurrentUserEntity.attribute\\_map \(nipyapi.nifi.models.labels\\_entity.LabelsEntity attribute\), 284](#)  
[attribute\\_map \(nipyapi.nifi.models.difference\\_dto.DifferenceDTO.to\\_map \(nipyapi.nifi.models.lineage\\_dto.LineageDTO attribute\), 286](#)  
[attribute\\_map \(nipyapi.nifi.models.dimensions\\_dto.DimensionsDTO.attribute\\_map \(nipyapi.nifi.models.lineage\\_entity.LineageEntity attribute\), 286](#)  
[attribute\\_map \(nipyapi.nifi.models.documented\\_type\\_dto.DocumentedTypeDTO.models.lineage\\_request\\_dto.LineageRequestDTO attribute\), 287](#)

`attribute_map(nipyapi.nifi.models.lineage_results_dto.LineageResultsDTO(nipyapi.nifi.models.previous_value_dto.PreviousValueDTO(attribute), 319 attribute), 348`  
`attribute_map(nipyapi.nifi.models.listing_request_dto.ListingRequestDTO(nipyapi.nifi.models.prioritizer_types_entity.PrioritizerTypesEntity(attribute), 320 attribute), 348`  
`attribute_map(nipyapi.nifi.models.listing_request_entity.ListingRequestEntity(nipyapi.nifi.models.process_group_dto.ProcessGroupDTO(attribute), 322 attribute), 351`  
`attribute_map(nipyapi.nifi.models.node_connection_status_snapshot_dto.NodeConnectionStatusSnapshotDTO(nipyapi.nifi.models.process_group_flow_dto.ProcessGroupFlowDTO(attribute), 323 attribute), 357`  
`attribute_map(nipyapi.nifi.models.node_counters_snapshot_dto.NodeCountersSnapshotDTO(nipyapi.nifi.models.process_group_flow_entity.ProcessGroupFlowEntity(attribute), 324 attribute), 361`  
`attribute_map(nipyapi.nifi.models.node_dto.NodeDTO(attribute_map(nipyapi.nifi.models.process_group_flow_entity.ProcessGroupFlowEntity(attribute), 325 attribute), 362`  
`attribute_map(nipyapi.nifi.models.node_entity.NodeEntity(attribute_map(nipyapi.nifi.models.process_group_status_dto.ProcessGroupStatusDTO(attribute), 326 attribute), 363`  
`attribute_map(nipyapi.nifi.models.node_event_dto.NodeEventDTO(attribute_map(nipyapi.nifi.models.process_group_status_entity.ProcessGroupStatusEntity(attribute), 327 attribute), 364`  
`attribute_map(nipyapi.nifi.models.node_port_status_snapshot_dto.NodePortStatusSnapshotDTO(nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO(attribute), 328 attribute), 366`  
`attribute_map(nipyapi.nifi.models.node_process_group_status_snapshot_dto.NodeProcessGroupStatusSnapshotDTO(nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO(attribute), 329 attribute), 370`  
`attribute_map(nipyapi.nifi.models.node_processor_status_snapshot_dto.NodeProcessorStatusSnapshotDTO(nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity(attribute), 330 attribute), 371`  
`attribute_map(nipyapi.nifi.models.node_remote_process_group_status_snapshot_dto.NodeRemoteProcessGroupStatusSnapshotDTO(nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity(attribute), 331 attribute), 372`  
`attribute_map(nipyapi.nifi.models.node_search_result_dto.NodeSearchResultDTO(nipyapi.nifi.models.processor_dto.ProcessorDTO(attribute), 332 attribute), 376`  
`attribute_map(nipyapi.nifi.models.node_status_snapshot_dto.NodeStatusSnapshotDTO(nipyapi.nifi.models.processor_entity.ProcessorEntity(attribute), 333 attribute), 379`  
`attribute_map(nipyapi.nifi.models.node_system_diagnostics_snapshot_dto.NodeSystemDiagnosticsSnapshotDTO(nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity(attribute), 334 attribute), 381`  
`attribute_map(nipyapi.nifi.models.output_ports_entity.OutputPortsEntity(attribute_map(nipyapi.nifi.models.processor_status_entity.ProcessorStatusEntity(attribute), 334 attribute), 383`  
`attribute_map(nipyapi.nifi.models.peer_dto.PeerDTO(attribute_map(nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO(attribute), 335 attribute), 384`  
`attribute_map(nipyapi.nifi.models.peers_entity.PeersEntity(attribute_map(nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity(attribute), 336 attribute), 387`  
`attribute_map(nipyapi.nifi.models.permissions.Permissions(attribute_map(nipyapi.nifi.models.processor_types_entity.ProcessorTypesEntity(attribute), 336 attribute), 388`  
`attribute_map(nipyapi.nifi.models.permissions_dto.PermissionsDTO(attribute_map(nipyapi.nifi.models.processors_entity.ProcessorsEntity(attribute), 337 attribute), 389`  
`attribute_map(nipyapi.nifi.models.port_dto.PortDTO(attribute_map(nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO(attribute), 338 attribute), 390`  
`attribute_map(nipyapi.nifi.models.port_entity.PortEntity(attribute_map(nipyapi.nifi.models.property_descriptor_entity.PropertyDescriptorEntity(attribute), 340 attribute), 392`  
`attribute_map(nipyapi.nifi.models.port_status_dto.PortStatusDTO(attribute_map(nipyapi.nifi.models.property_history_dto.PropertyHistoryDTO(attribute), 342 attribute), 393`  
`attribute_map(nipyapi.nifi.models.port_status_entity.PortStatusEntity(attribute_map(nipyapi.nifi.models.provenance_dto.ProvenanceDTO(attribute), 343 attribute), 393`  
`attribute_map(nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO(nipyapi.nifi.models.provenance_entity.ProvenanceEntity(attribute), 344 attribute), 395`  
`attribute_map(nipyapi.nifi.models.port_status_snapshot_entity.PortStatusSnapshotEntity(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO(attribute), 346 attribute), 397`  
`attribute_map(nipyapi.nifi.models.position_dto.PositionDTO(attribute_map(nipyapi.nifi.models.provenance_event_entity.ProvenanceEventEntity(attribute), 347 attribute), 402`

attribute\_map (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO\_entity.ResourcesEntity  
 attribute), 402 attribute), 442  
 attribute\_map (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO\_entity.RevisionEntity  
 attribute), 404 attribute), 443  
 attribute\_map (nipyapi.nifi.models.provenance\_options\_dto.ProvenanceOptionsDTO\_entity.ScheduleComponentsEntity  
 attribute), 405 attribute), 444  
 attribute\_map (nipyapi.nifi.models.provenance\_options\_entity.ProvenanceOptionsEntity\_entity.SearchResultsEntity  
 attribute), 406 attribute), 445  
 attribute\_map (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO\_entity.SearchResultsEntity  
 attribute), 407 attribute), 447  
 attribute\_map (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO\_entity.SnippetEntity  
 attribute), 409 attribute), 447  
 attribute\_map (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO\_entity.SnippetEntity  
 attribute), 410 attribute), 449  
 attribute\_map (nipyapi.nifi.models.queue\_size\_dto.QueueSizeDTO\_entity.StateEntryEntity  
 attribute), 411 attribute), 450  
 attribute\_map (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity\_entity.StateEntryEntity  
 attribute), 412 attribute), 451  
 attribute\_map (nipyapi.nifi.models.registry\_clients\_entity.RegistryClientsEntity\_entity.StateMapEntity  
 attribute), 413 attribute), 452  
 attribute\_map (nipyapi.nifi.models.registry\_dto.RegistryDTO\_entity.StateMapEntity  
 attribute), 414 attribute), 453  
 attribute\_map (nipyapi.nifi.models.relationship\_dto.RelationshipDTO\_entity.StatusHistoryEntity  
 attribute), 415 attribute), 454  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_contents\_dto.RemoteProcessGroupContentsDTO\_entity.StatusHistoryEntity  
 attribute), 416 attribute), 455  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO\_entity.StatusSnapshotEntity  
 attribute), 418 attribute), 455  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity\_entity.StorageUsageEntity  
 attribute), 422 attribute), 456  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_path\_dto.RemoteProcessGroupPathDTO\_entity.StreamingOutputEntity  
 attribute), 424 attribute), 458  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_path\_entity.RemoteProcessGroupPathEntity\_entity.SubmissionRequestEntity  
 attribute), 426 attribute), 458  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO\_entity.SystemDiagnosticsEntity  
 attribute), 428 attribute), 459  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity\_entity.SystemDiagnosticsEntity  
 attribute), 430 attribute), 459  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshotDTO\_entity.StatusSnapshotEntity  
 attribute), 431 attribute), 461  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity.RemoteProcessGroupStatusSnapshotEntity\_entity.StatusSnapshotEntity  
 attribute), 433 attribute), 465  
 attribute\_map (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity.RemoteProcessGroupStatusSnapshotEntity\_entity.TemplateEntity  
 attribute), 434 attribute), 467  
 attribute\_map (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO\_entity.TemplatesEntity  
 attribute), 436 attribute), 468  
 attribute\_map (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity\_entity.TenantEntity  
 attribute), 439 attribute), 469  
 attribute\_map (nipyapi.nifi.models.reporting\_task\_type\_entity.ReportingTaskTypeEntity\_entity.TenantEntity  
 attribute), 440 attribute), 470  
 attribute\_map (nipyapi.nifi.models.reporting\_tasks\_entity.ReportingTasksEntity\_entity.TenantsEntity  
 attribute), 441 attribute), 471  
 attribute\_map (nipyapi.nifi.models.resource\_dto.ResourceDTO\_entity.TransactionResultEntity  
 attribute), 442 attribute), 472

[attribute\\_map \(nipyapi.nifi.models.update\\_controller\\_service\\_reference\\_request.update\\_controller\\_service\\_reference\\_request\\_attribute\), 473](#)  
[attribute\\_map \(nipyapi.nifi.models.user\\_dto.UserDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_update\\_request.dto.attribute\), 474](#)  
[attribute\\_map \(nipyapi.nifi.models.user\\_entity.UserEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_update\\_request.entity.attribute\), 475](#)  
[attribute\\_map \(nipyapi.nifi.models.user\\_group\\_dto.UserGroupDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_flows\\_entity.VersionedFlowsEntity.attribute\), 477](#)  
[attribute\\_map \(nipyapi.nifi.models.user\\_group\\_entity.UserGroupEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_funnel.VersionedFunnel.attribute\), 478](#)  
[attribute\\_map \(nipyapi.nifi.models.user\\_groups\\_entity.UserGroupsEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_label.VersionedLabel.attribute\), 480](#)  
[attribute\\_map \(nipyapi.nifi.models.users\\_entity.UsersEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_port.VersionedPort.attribute\), 480](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_dto.VariableDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_process\\_group.VersionedProcessGroup.attribute\), 481](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_entity.VariableEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_processor.VersionedProcessor.attribute\), 482](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_registry\\_dto.VariableRegistryDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_property\\_descriptor.VersionedPropertyDescriptor.attribute\), 482](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_registry\\_entity.VariableRegistryEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort.attribute\), 483](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_registry\\_update\\_request.dto.VariableRegistryUpdateRequestDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort.attribute\), 484](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_registry\\_update\\_request.entity.VariableRegistryUpdateRequestEntity.attribute\\_map \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort.attribute\), 486](#)  
[attribute\\_map \(nipyapi.nifi.models.variable\\_registry\\_update\\_step\\_dto.VariableRegistryUpdateStepDTO.attribute\\_map \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort.attribute\), 487](#)  
[attribute\\_map \(nipyapi.nifi.models.version\\_control\\_component\\_wrapping\\_entity.VersionControlComponentWrappingEntity.attribute\\_map \(nipyapi.nifi.models.version\\_control\\_component\\_wrapping\\_entity.VersionControlComponentWrappingEntity.attribute\), 488](#)  
[attribute\\_map \(nipyapi.nifi.models.version\\_control\\_information\\_dto.VersionControlInformationDTO.attribute\\_map \(nipyapi.nifi.models.version\\_control\\_information\\_entity.VersionControlInformationEntity.attribute\), 489](#)  
[attribute\\_map \(nipyapi.nifi.models.version\\_control\\_information\\_entity.VersionControlInformationEntity.attribute\\_map \(nipyapi.nifi.models.version\\_control\\_information\\_entity.VersionControlInformationEntity.attribute\), 491](#)  
[attribute\\_map \(nipyapi.nifi.models.version\\_info\\_dto.VersionInfoDTO.attribute\\_map \(nipyapi.registry.models.bundle.Bundle.attribute\), 492](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_connection.VersionedConnection.attribute\\_map \(nipyapi.registry.models.connectable\\_component.ConnectableComponent.attribute\), 494](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_controller\\_service.VersionedControllerService.attribute\\_map \(nipyapi.registry.models.controller\\_service\\_api.ControllerServiceAPI.attribute\), 498](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow.VersionedFlow.attribute\\_map \(nipyapi.registry.models.current\\_user.CurrentUser.attribute\), 500](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_coordinates.VersionedFlowCoordinates.attribute\\_map \(nipyapi.registry.models.fields.Fields.attribute\), 502](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_dto.VersionedFlowDTO.attribute\\_map \(nipyapi.registry.models.permissions.Permissions.attribute\), 503](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_entity.VersionedFlowEntity.attribute\\_map \(nipyapi.registry.models.resource.Resource.attribute\), 504](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_snapshot.VersionedFlowSnapshot.attribute\\_map \(nipyapi.registry.models.resource\\_permissions.ResourcePermissions.attribute\), 505](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_identity.VersionedFlowSnapshotIdentity.attribute\\_map \(nipyapi.registry.models.tenant.Tenant.attribute\), 506](#)  
[attribute\\_map \(nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_metadata\\_entity.VersionedFlowSnapshotMetadataEntity.attribute\\_map \(nipyapi.registry.models.versioned\\_flow\\_snapshot\\_metadata\\_entity.VersionedFlowSnapshotMetadataEntity.attribute\), 507](#)



[attribute\\_map \(nipyapi.registry.models.user\\_group.UserGroup attribute\), 609](#)  
[attribute\), 584](#) [available\\_processors](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_connection.VersionedConnection attribute\), 586](#) [system\\_diagnostics\\_snapshot\\_dto.SystemDiagnosticsSnapshotDTO attribute\), 461](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_controller\\_service.VersionedControllerService attribute\), 590](#) [\(nipyapi.nifi.models.connection\\_dto.ConnectionDTO attribute\), 239](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_flow.VersionedFlow attribute\), 592](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_flow\\_coordinates.VersionedFlowCoordinates attribute\), 594](#) [back\\_pressure\\_data\\_size\\_threshold](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_flow\\_snapshot.VersionedFlowSnapshot attribute\), 596](#) [\(nipyapi.nifi.models.connection\\_dto.ConnectionDTO attribute\), 239](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_flow\\_snapshot\\_metadata.VersionedFlowSnapshotMetadata attribute\), 597](#) [\(nipyapi.nifi.models.versioned\\_connection.VersionedConnection attribute\), 494](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_funnel.VersionedFunnel attribute\), 598](#) [back\\_pressure\\_data\\_size\\_threshold](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_label.VersionedLabel attribute\), 600](#) [\(nipyapi.registry.models.versioned\\_connection.VersionedConnection attribute\), 586](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_port.VersionedPort attribute\), 602](#) [back\\_pressure\\_object\\_threshold](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_process\\_group.VersionedProcessGroup attribute\), 604](#) [\(nipyapi.nifi.models.connection\\_dto.ConnectionDTO attribute\), 239](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_processor.VersionedProcessor attribute\), 609](#) [back\\_pressure\\_object\\_threshold](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_processor\\_models.versioned\\_connection.VersionedConnection attribute\), 494](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_property\\_descriptor.VersionedPropertyDescriptor attribute\), 612](#) [\(nipyapi.registry.models.versioned\\_connection.VersionedConnection attribute\), 586](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\), 613](#) [backoff\\_mechanism](#)  
[attribute\\_map \(nipyapi.registry.models.versioned\\_remote\\_process\\_group.VersionedRemoteProcessGroup attribute\), 616](#) [\(nipyapi.nifi.models.processor\\_config\\_dto.ProcessorConfigDTO attribute\), 373](#)  
[AttributeDTO \(class in backoff\\_mechanism](#)  
[nipyapi.nifi.models.attribute\\_dto\), 211](#) [\(nipyapi.nifi.models.versioned\\_processor.VersionedProcessor attribute\), 523](#)  
[attributes \(nipyapi.nifi.models.flow\\_file\\_dto.FlowFileDTO attribute\), 299](#) [backoff\\_mechanism](#)  
[attributes \(nipyapi.nifi.models.provenance\\_event\\_dto.ProvenanceEventDTO attribute\), 397](#) [\(nipyapi.registry.models.versioned\\_processor.VersionedProcessor attribute\), 609](#)  
[author \(nipyapi.registry.models.versioned\\_flow\\_snapshot\\_metadata.VersionedFlowSnapshotMetadata attribute\), 597](#) [BannerEntity \(class in nipyapi.nifi.models.banner\\_dto\), 211](#)  
[authorization\\_issues](#) [BannerEntity \(class in nipyapi.nifi.models.banner\\_entity\), 212](#)  
[\(nipyapi.nifi.models.remote\\_process\\_group\\_dto.RemoteProcessGroupDTO attribute\), 418](#) [banners \(nipyapi.nifi.models.banner\\_entity.BannerEntity attribute\), 212](#)  
[auto\\_refresh\\_interval\\_seconds](#) [\(nipyapi.nifi.models.flow\\_configuration\\_dto.FlowConfigurationDTO attribute\), 295](#) [batch\\_settings \(nipyapi.nifi.models.remote\\_process\\_group\\_port\\_dto.RemoteProcessGroupPortDTO attribute\), 424](#)  
[auto\\_terminate \(nipyapi.nifi.models.relationship\\_dto.RelationshipDTO attribute\), 415](#) [\(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\), 528](#)  
[auto\\_terminated\\_relationships](#) [batch\\_size \(nipyapi.registry.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\), 614](#)  
[\(nipyapi.nifi.models.processor\\_config\\_dto.ProcessorConfigDTO attribute\), 372](#) [BatchSettingsDTO \(class in nipyapi.nifi.models.batch\\_settings\\_dto\), 213](#)  
[auto\\_terminated\\_relationships](#) [batch\\_size \(class in nipyapi.nifi.models.batch\\_size\), 213](#)  
[\(nipyapi.nifi.models.versioned\\_processor.VersionedProcessor attribute\), 523](#) [BatchSize \(class in nipyapi.registry.models.batch\\_size\), 571](#)  
[auto\\_terminated\\_relationships](#) [BatchSize \(class in nipyapi.registry.models.batch\\_size\), 571](#)

bends (*nipyapi.nifi.models.connection\_dto.ConnectionDTO* attribute), 239      *BucketFlowsApi* (class in *nipyapi.registry.apis.bucket\_flows\_api*), 543  
 bends (*nipyapi.nifi.models.connection\_entity.ConnectionEntity* attribute), 242      *BucketItem* (class in *nipyapi.registry.models.bucket\_item*), 573  
 bends (*nipyapi.nifi.models.versioned\_connection.VersionedConnection* attribute), 494      (*nipyapi.nifi.models.buckets\_entity.BucketsEntity* attribute), 217  
 bends (*nipyapi.registry.models.versioned\_connection.VersionedConnection* attribute), 587      (*nipyapi.registry.models.resource\_permissions.ResourcePermissions* attribute), 581  
 bootstrap\_security\_policies() (in module *BucketsApi* (class in *nipyapi.security*), 24      (*nipyapi.registry.apis.buckets\_api*), 551  
 breadcrumb (*nipyapi.nifi.models.flow\_breadcrumb\_entity.FlowBreadcrumbEntity* attribute), 293      (class in *nipyapi.nifi.models.buckets\_entity*), 217  
 breadcrumb (*nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO* attribute), 361      (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 Bucket (class in *nipyapi.nifi.models.bucket*), 214      build\_branch (*nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO* attribute), 492  
 Bucket (class in *nipyapi.registry.models.bucket*), 571      build\_revision (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 bucket (*nipyapi.nifi.models.bucket\_entity.BucketEntity* attribute), 217      (*nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO* attribute), 492  
 bucket (*nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot* attribute), 505      (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 bucket (*nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot* attribute), 596      (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 bucket\_id (*nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO* attribute), 489      (*nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO* attribute), 492  
 bucket\_id (*nipyapi.nifi.models.versioned\_flow\_coordinates.VersionedFlowCoordinates* attribute), 502      (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 bucket\_id (*nipyapi.nifi.models.versioned\_flow\_dto.VersionedFlowDTO* attribute), 503      (*nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO* attribute), 492  
 bucket\_id (*nipyapi.registry.models.versioned\_flow\_coordinates.VersionedFlowCoordinates* attribute), 594      (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 bucket\_identifier      bulletin\_board (*nipyapi.nifi.models.bulletin\_board\_entity.BulletinBoardEntity* attribute), 219  
     (*nipyapi.nifi.models.versioned\_flow.VersionedFlow* attribute), 500      bulletin\_level (*nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO* attribute), 262  
 bucket\_identifier      bulletin\_level (*nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO* attribute), 373  
     (*nipyapi.registry.models.bucket\_item.BucketItem* attribute), 573      bulletin\_level (*nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService* attribute), 498  
 bucket\_identifier      bulletin\_level (*nipyapi.nifi.models.versioned\_processor.VersionedProcessor* attribute), 523  
     (*nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot* attribute), 597      (*nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService* attribute), 590  
 bucket\_name (*nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO* attribute), 489      (*nipyapi.nifi.models.versioned\_processor.VersionedProcessor* attribute), 609  
 bucket\_name (*nipyapi.nifi.models.versioned\_flow.VersionedFlow* attribute), 500      (*nipyapi.nifi.models.bulletin\_board\_dto*), 218  
 bucket\_name (*nipyapi.registry.models.bucket\_item.BucketItem* attribute), 573      *BulletinBoardEntity* (class in *nipyapi.nifi.models.bulletin\_board\_entity*), 219  
 bucket\_name (*nipyapi.registry.models.versioned\_flow.VersionedFlow* attribute), 593  
 BucketDTO (class in *nipyapi.nifi.models.bucket\_dto*), 216      *BulletinDTO* (class in *nipyapi.nifi.models.bulletin\_dto*), 219  
 BucketEntity (class in *nipyapi.nifi.models.bucket\_entity*), 217      *BulletinEntity* (class in *nipyapi.nifi.models.bulletin\_entity*), 221

bulletins (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 196 attribute), 376

bulletins (nipyapi.nifi.models.access\_policy\_summary\_entity.AccessPolicySummaryEntity (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 199 attribute), 436

bulletins (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService attribute), 208 attribute), 498

bulletins (nipyapi.nifi.models.bulletin\_board\_dto.BulletinBoardDTO (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 218 attribute), 523

bulletins (nipyapi.nifi.models.component\_reference\_entity.ComponentReferenceEntity (nipyapi.nifi.models.controller\_service\_api.ControllerServiceAPI attribute), 232 attribute), 577

bulletins (nipyapi.nifi.models.connection\_entity.ConnectionEntity (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService attribute), 242 attribute), 590

bulletins (nipyapi.nifi.models.controller\_bulletins\_entity.ControllerBulletinsEntity (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 253 attribute), 609

bulletins (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity (nipyapi.nifi.models.bundle\_dto), attribute), 265 223

bulletins (nipyapi.nifi.models.controller\_service\_reference\_entity.ControllerServiceReferenceEntity (nipyapi.nifi.models.component\_entity.ComponentEntity attribute), 270 attribute), 30

bulletins (nipyapi.nifi.models.funnel\_entity.FunnelEntity (nipyapi.nifi.models.queue\_size\_dto.QueueSizeDTO attribute), 306 attribute), 411

bulletins (nipyapi.nifi.models.label\_entity.LabelEntity (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshot attribute), 314 attribute), 248

bulletins (nipyapi.nifi.models.port\_entity.PortEntity (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshot attribute), 340 attribute), 344

bulletins (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 357 attribute), 366

bulletins (nipyapi.nifi.models.processor\_entity.ProcessorEntity (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshot attribute), 379 attribute), 384

bulletins (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshot attribute), 412 attribute), 248

bulletins (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshot attribute), 422 attribute), 344

bulletins (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 427 attribute), 366

bulletins (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshot attribute), 439 attribute), 384

bulletins (nipyapi.nifi.models.template\_entity.TemplateEntity (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshot attribute), 467 attribute), 248

bulletins (nipyapi.nifi.models.tenant\_entity.TenantEntity (nipyapi.nifi.models.controller\_status\_dto.ControllerStatus attribute), 470 attribute), 274

bulletins (nipyapi.nifi.models.user\_entity.UserEntity (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 475 attribute), 366

bulletins (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 478 attribute), 366

Bundle (class in nipyapi.nifi.models.bundle), 222 bytes\_read (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshot attribute), 385

Bundle (class in nipyapi.registry.models.bundle), 575 attribute), 385

bundle (nipyapi.nifi.models.controller\_service\_api.ControllerServiceAPI (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 259 attribute), 366

bundle (nipyapi.nifi.models.controller\_service\_api\_dto.ControllerServiceAPIDTO (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshot attribute), 260 attribute), 431

bundle (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshot attribute), 262 attribute), 366

bundle (nipyapi.nifi.models.documented\_type\_dto.DocumentedTypeDTO (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshot attribute), 287 attribute), 431

bytes\_transferred (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 366  
 bytes\_written (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 367  
 bytes\_written (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 385

## C

call\_api () (nipyapi.nifi.api\_client.ApiClient method), 533  
 call\_api () (nipyapi.registry.api\_client.ApiClient method), 619  
 can\_delete (nipyapi.nifi.models.permissions.Permissions attribute), 336  
 can\_delete (nipyapi.registry.models.permissions.Permissions attribute), 579  
 can\_read (nipyapi.nifi.models.action\_entity.ActionEntity attribute), 204  
 can\_read (nipyapi.nifi.models.allowable\_value\_entity.AllowableValueEntity attribute), 210  
 can\_read (nipyapi.nifi.models.bulletin\_entity.BulletinEntity attribute), 221  
 can\_read (nipyapi.nifi.models.connection\_status\_entity.ConnectionStatusEntity attribute), 246  
 can\_read (nipyapi.nifi.models.connection\_status\_snapshot\_entity.ConnectionStatusSnapshotEntity attribute), 251  
 can\_read (nipyapi.nifi.models.permissions.Permissions attribute), 337  
 can\_read (nipyapi.nifi.models.permissions\_dto.PermissionsDTO attribute), 337  
 can\_read (nipyapi.nifi.models.port\_status\_entity.PortStatusEntity attribute), 343  
 can\_read (nipyapi.nifi.models.port\_status\_snapshot\_entity.PortStatusSnapshotEntity attribute), 346  
 can\_read (nipyapi.nifi.models.process\_group\_status\_entity.ProcessGroupStatusEntity attribute), 364  
 can\_read (nipyapi.nifi.models.process\_group\_status\_snapshot\_entity.ProcessGroupStatusSnapshotEntity attribute), 370  
 can\_read (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity attribute), 383  
 can\_read (nipyapi.nifi.models.processor\_status\_snapshot\_entity.ProcessorStatusSnapshotEntity attribute), 387  
 can\_read (nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity attribute), 430  
 can\_read (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity.RemoteProcessGroupStatusSnapshotEntity attribute), 433  
 can\_read (nipyapi.nifi.models.status\_history\_entity.StatusHistoryEntity attribute), 455  
 can\_read (nipyapi.registry.models.permissions.Permissions attribute), 579  
 can\_version\_flows (nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity attribute), 284  
 can\_write (nipyapi.nifi.models.permissions.Permissions attribute), 337  
 can\_write (nipyapi.nifi.models.permissions\_dto.PermissionsDTO attribute), 337  
 can\_write (nipyapi.nifi.models.variable\_entity.VariableEntity attribute), 483  
 can\_write (nipyapi.registry.models.permissions.Permissions attribute), 579  
 category (nipyapi.nifi.models.bulletin\_dto.BulletinDTO attribute), 219  
 category (nipyapi.nifi.models.node\_event\_dto.NodeEventDTO attribute), 327  
 check\_version () (in module nipyapi.utils), 30  
 child\_uids (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 397  
 child\_uids (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO attribute), 404  
 clear\_state () (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi method), 52  
 clear\_state () (nipyapi.nifi.apis.processors\_api.ProcessorsApi method), 141  
 clear\_state () (nipyapi.nifi.apis.reporting\_tasks\_api.ReportingTasksApi method), 163  
 clear\_state\_with\_http\_info () (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi method), 53  
 clear\_state\_with\_http\_info () (nipyapi.nifi.apis.processors\_api.ProcessorsApi method), 141  
 clear\_state\_with\_http\_info () (nipyapi.nifi.apis.reporting\_tasks\_api.ReportingTasksApi method), 163  
 client\_id (nipyapi.nifi.models.revision\_dto.RevisionDTO attribute), 443  
 cluster (nipyapi.nifi.models.cluster\_entity.ClusterEntity attribute), 225  
 cluster\_node\_address (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 299  
 cluster\_node\_address (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 303  
 cluster\_node\_address (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 397  
 cluster\_node\_address (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity.RemoteProcessGroupStatusSnapshotEntity attribute), 451  
 cluster\_node\_id (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 299  
 cluster\_node\_id (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 302  
 cluster\_node\_id (nipyapi.nifi.models.lineage\_request\_dto.LineageRequestDTO attribute), 318



`cluster_node_id(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute), 397` `comments(nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO`  
`cluster_node_id(nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDTO`  
`attribute), 407` `comments(nipyapi.nifi.models.versioned_connection.VersionedConnection`  
`cluster_node_id(nipyapi.nifi.models.state_entry_dto.StateEntryDTO), 495`  
`attribute), 451` `comments(nipyapi.nifi.models.versioned_controller_service.VersionedCo`  
`cluster_node_id(nipyapi.nifi.models.submit_replay_request_entity.SubmitReplayRequestEntity`  
`attribute), 458` `comments(nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO`  
`cluster_node_identifier` `attribute), 503`  
`(nipyapi.nifi.models.provenance_node_dto.ProvenanceNodeDTO), 404` `comments(nipyapi.nifi.models.versioned_funnel.VersionedFunnel`  
`attribute), 512`  
`cluster_state(nipyapi.nifi.models.component_state_dto.ComponentStateDTO), 513`  
`attribute), 234` `comments(nipyapi.nifi.models.versioned_label.VersionedLabel`  
`cluster_summary(nipyapi.nifi.models.cluste_summary_entity.ClusterSummaryEntity), 515`  
`attribute), 224` `attribute), 515`  
`ClusterDTO (class in nipyapi.nifi.models.cluster_dto),` `comments(nipyapi.nifi.models.versioned_process_group.VersionedProces`  
`224` `attribute), 519`  
`clustered(nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO), 523`  
`attribute), 226` `attribute), 523`  
`ClusterEntity (class in` `comments(nipyapi.nifi.models.versioned_remote_group_port.VersionedR`  
`nipyapi.nifi.models.cluster_entity), 225` `attribute), 528`  
`ClusterSearchResultsEntity (class in` `comments(nipyapi.nifi.models.versioned_remote_process_group.Versione`  
`nipyapi.nifi.models.cluster_search_results_entity),` `attribute), 530`  
`225` `comments(nipyapi.registry.models.connectable_component.ConnectableC`  
`ClusterSummaryDTO (class in` `attribute), 576`  
`nipyapi.nifi.models.cluster_summary_dto),` `comments(nipyapi.registry.models.versioned_connection.VersionedConne`  
`226` `attribute), 587`  
`ClusteSummaryEntity (class in` `comments(nipyapi.registry.models.versioned_controller_service.Versione`  
`nipyapi.nifi.models.cluste_summary_entity),` `attribute), 591`  
`223` `comments(nipyapi.registry.models.versioned_flow_snapshot_metadata.Ve`  
`collection_count(nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO`  
`attribute), 308` `comments(nipyapi.registry.models.versioned_funnel.VersionedFunnel`  
`collection_millis` `attribute), 599`  
`(nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO), 309` `comments(nipyapi.registry.models.versioned_label.VersionedLabel`  
`attribute), 309` `attribute), 600`  
`collection_time(nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO), 309` `comments(nipyapi.registry.models.versioned_port.VersionedPort`  
`attribute), 309` `attribute), 602`  
`comments(nipyapi.nifi.models.connectable_component.ConnectableComponent), 605`  
`attribute), 236` `attribute), 605`  
`comments(nipyapi.nifi.models.connectable_dto.ConnectableDTO), 609`  
`attribute), 237` `attribute), 609`  
`comments(nipyapi.nifi.models.controller_dto.ControllerDTO), 614`  
`attribute), 256` `attribute), 614`  
`comments(nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO), 616`  
`attribute), 262` `attribute), 616`  
`comments(nipyapi.nifi.models.port_dto.PortDTO` `commit_input_port_transaction()`  
`attribute), 338` `(nipyapi.nifi.apis.data_transfer_api.DataTransferApi`  
`comments(nipyapi.nifi.models.process_group_dto.ProcessGroupDTO), 61`  
`attribute), 351` `commit_input_port_transaction_with_http_info()`  
`comments(nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO), 61`  
`attribute), 373` `(nipyapi.nifi.apis.data_transfer_api.DataTransferApi`  
`comments(nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO), 61`  
`attribute), 418` `method), 61`  
`comments(nipyapi.nifi.models.remote_process_group_port_dto.RemoteProcessGroupPortDTO`  
`attribute), 418` `method), 61`

commit\_output\_port\_transaction\_with\_httpcomponent (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi attribute), 478  
method), 62 component\_details

communications\_timeout (nipyapi.nifi.models.action\_dto.ActionDTO (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 418 component\_details

communications\_timeout (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 530 component\_differences

communications\_timeout (nipyapi.nifi.models.flow\_comparison\_entity.FlowComparisonEntity (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 616 component\_history

complete (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO (nipyapi.nifi.models.variable\_registry\_entity.ComponentHistory attribute), 484 attribute), 229

complete (nipyapi.nifi.models.variable\_registry\_update\_step\_dto.VariableRegistryUpdateStepDTO (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO attribute), 487 attribute), 228

complete (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO (nipyapi.nifi.models.component\_history\_dto.ComponentHistory attribute), 509 attribute), 229

component (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity id (nipyapi.nifi.models.component\_state\_dto.ComponentStateDTO attribute), 196 attribute), 234

component (nipyapi.nifi.models.access\_policy\_summary\_entity.AccessPolicySummaryEntity (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 199 attribute), 397

component (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO attribute), 208 attribute), 228

component (nipyapi.nifi.models.component\_reference\_entity.ComponentReferenceEntity (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 232 attribute), 397

component (nipyapi.nifi.models.connection\_entity.ConnectionEntity connection\_reference (nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO attribute), 242 attribute), 194

component (nipyapi.nifi.models.controller\_configuration\_entity.ControllerConfigurationEntity controller\_configuration (nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO attribute), 254 attribute), 198

component (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity (nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO attribute), 265 attribute), 198

component (nipyapi.nifi.models.controller\_service\_referencing\_component\_entity.ControllerServiceReferencingComponentEntity (nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity attribute), 270 attribute), 284

component (nipyapi.nifi.models.funnel\_entity.FunnelEntity component\_state (nipyapi.nifi.models.component\_state\_entity.ComponentStateEntity attribute), 306 attribute), 235

component (nipyapi.nifi.models.label\_entity.LabelEntity component\_type (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO attribute), 314 attribute), 228

component (nipyapi.nifi.models.port\_entity.PortEntity component\_type (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 340 attribute), 397

component (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 357 attribute), 495

component (nipyapi.nifi.models.processor\_entity.ProcessorEntity component\_type (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService attribute), 380 attribute), 498

component (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity (nipyapi.nifi.models.versioned\_funnel.VersionedFunnel attribute), 412 attribute), 513

component (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity (nipyapi.nifi.models.versioned\_label.VersionedLabel attribute), 422 attribute), 515

component (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity (nipyapi.nifi.models.versioned\_port.VersionedPort attribute), 439 attribute), 519

component (nipyapi.nifi.models.tenant\_entity.TenantEntity component\_type (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 470 attribute), 519

component (nipyapi.nifi.models.user\_entity.UserEntity component\_type (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 475 attribute), 519

[attribute](#)), 523  
[component\\_type \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPortDTO attribute\)](#), 528  
[component\\_type \(nipyapi.nifi.models.versioned\\_remote\\_processor\\_group.VersionedRemoteProcessorGroup attribute\)](#), 530  
[component\\_type \(nipyapi.registry.models.versioned\\_connection.VersionedConnection attribute\)](#), 587  
[component\\_type \(nipyapi.registry.models.versioned\\_controller\\_service.VersionedControllerService attribute\)](#), 591  
[component\\_type \(nipyapi.registry.models.versioned\\_funnel.VersionedFunnel attribute\)](#), 599  
[component\\_type \(nipyapi.registry.models.versioned\\_label.VersionedLabel attribute\)](#), 600  
[component\\_type \(nipyapi.registry.models.versioned\\_port.VersionedPort attribute\)](#), 602  
[component\\_type \(nipyapi.registry.models.versioned\\_processor\\_group.VersionedProcessorGroup attribute\)](#), 605  
[component\\_type \(nipyapi.registry.models.versioned\\_processor.VersionedProcessor attribute\)](#), 609  
[component\\_type \(nipyapi.registry.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\)](#), 614  
[component\\_type \(nipyapi.registry.models.versioned\\_remote\\_processor\\_group.VersionedRemoteProcessorGroup attribute\)](#), 616  
[ComponentDetailsDTO \(class in nipyapi.nifi.models.component\\_details\\_dto\)](#), 227  
[ComponentDifferenceDTO \(class in nipyapi.nifi.models.component\\_difference\\_dto\)](#), 227  
[ComponentHistoryDTO \(class in nipyapi.nifi.models.component\\_history\\_dto\)](#), 229  
[ComponentHistoryEntity \(class in nipyapi.nifi.models.component\\_history\\_entity\)](#), 229  
[ComponentReferenceDTO \(class in nipyapi.nifi.models.component\\_reference\\_dto\)](#), 230  
[ComponentReferenceEntity \(class in nipyapi.nifi.models.component\\_reference\\_entity\)](#), 231  
[components \(nipyapi.nifi.models.activate\\_controller\\_services\\_entity.ActiveControllerServicesEntity attribute\)](#), 205  
[components \(nipyapi.nifi.models.schedule\\_components\\_entity.ScheduleComponentsEntity attribute\)](#), 444  
[ComponentSearchResultDTO \(class in nipyapi.nifi.models.component\\_search\\_result\\_dto\)](#), 233  
[ComponentStateDTO \(class in nipyapi.nifi.models.component\\_state\\_dto\)](#), 234  
[ComponentStateEntity \(class in nipyapi.nifi.models.component\\_state\\_entity\)](#), 235  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\)](#), 338  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.processor\\_config\\_dto.ProcessorConfigDTO attribute\)](#), 377  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_remote\\_processor\\_group.VersionedRemoteProcessorGroup attribute\)](#), 424  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_port.VersionedPort attribute\)](#), 516  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_processor.VersionedProcessor attribute\)](#), 523  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\)](#), 528  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.nifi.models.versioned\\_remote\\_processor\\_group.VersionedRemoteProcessorGroup attribute\)](#), 602  
[concurrently\\_schedulable\\_task\\_count \(nipyapi.registry.models.versioned\\_remote\\_group\\_port.VersionedRemoteGroupPort attribute\)](#), 609  
[config \(nipyapi.nifi.models.access\\_configuration\\_entity.AccessConfigurationEntity attribute\)](#), 193  
[config \(nipyapi.nifi.models.processor\\_dto.ProcessorDTO attribute\)](#), 376  
[configurable \(nipyapi.nifi.models.access\\_policy\\_dto.AccessPolicyDTO attribute\)](#), 194  
[configurable \(nipyapi.nifi.models.access\\_policy\\_summary\\_dto.AccessPolicySummaryDTO attribute\)](#), 198  
[configurable \(nipyapi.nifi.models.tenant\\_dto.TenantDTO attribute\)](#), 469  
[configurable \(nipyapi.nifi.models.user\\_dto.UserDTO attribute\)](#), 474  
[configurable \(nipyapi.nifi.models.user\\_group\\_dto.UserGroupDTO attribute\)](#), 477  
[configurable \(nipyapi.registry.models.access\\_policy.AccessPolicy attribute\)](#), 568  
[configurable \(nipyapi.registry.models.access\\_policy\\_summary.AccessPolicySummary attribute\)](#), 570  
[configurable \(nipyapi.registry.models.tenant.Tenant attribute\)](#), 582  
[configurable \(nipyapi.registry.models.user.User attribute\)](#), 583  
[configurable \(nipyapi.registry.models.user\\_group.UserGroup attribute\)](#), 584  
[Configuration\(\) \(in module nipyapi.nifi.configuration\)](#), 535  
[Configuration\(\) \(in module nipyapi.nifi.configuration\)](#), 535

*nipyapi.registry.configuration*), 621  
 ConnectableComponent (class in *nipyapi.nifi.models.connectable\_component*), 236  
 ConnectableComponent (class in *nipyapi.registry.models.connectable\_component*), 575  
 ConnectableDTO (class in *nipyapi.nifi.models.connectable\_dto*), 237  
 connected (*nipyapi.nifi.models.remote\_process\_group\_processor\_remote\_process\_group\_processor\_dto* attribute), 425  
 connected\_node\_count (*nipyapi.nifi.models.cluster\_summary\_dto.ClusterSummaryDTO* attribute), 226  
 connected\_nodes (*nipyapi.nifi.models.cluster\_summary\_dto.ClusterSummaryDTO* attribute), 226  
 connected\_to\_cluster (*nipyapi.nifi.models.cluster\_summary\_dto.ClusterSummaryDTO* attribute), 226  
 connection\_requested (*nipyapi.nifi.models.node\_dto.NodeDTO* attribute), 325  
 connection\_results (*nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO* attribute), 445  
 connection\_status (*nipyapi.nifi.models.connection\_status\_entity.ConnectionStatusEntity* attribute), 246  
 connection\_status\_snapshot (*nipyapi.nifi.models.connection\_status\_snapshot\_entity.ConnectionStatusSnapshotEntity* attribute), 251  
 connection\_status\_snapshots (*nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO* attribute), 367  
 ConnectionDTO (class in *nipyapi.nifi.models.connection\_dto*), 238  
 ConnectionEntity (class in *nipyapi.nifi.models.connection\_entity*), 242  
 connections (*nipyapi.nifi.models.connections\_entity.ConnectionsEntity* attribute), 252  
 connections (*nipyapi.nifi.models.flow\_dto.FlowDTO* attribute), 297  
 connections (*nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO* attribute), 304  
 connections (*nipyapi.nifi.models.snippet\_dto.SnippetDTO* attribute), 447  
 connections (*nipyapi.nifi.models.versioned\_process\_group\_versioned\_process\_group* attribute), 519  
 connections (*nipyapi.registry.models.versioned\_process\_group\_versioned\_process\_group* attribute), 605  
 ConnectionsApi (class in *nipyapi.nifi.apis.connections\_api*), 41  
 ConnectionsEntity (class in *nipyapi.nifi.models.connections\_entity*), 252  
 ConnectionStatusDTO (class in *nipyapi.nifi.models.connection\_status\_dto*), 244  
 ConnectionStatusEntity (class in *nipyapi.nifi.models.connection\_status\_entity*), 246  
 ConnectionStatusSnapshotDTO (class in *nipyapi.nifi.models.connection\_status\_snapshot\_dto*), 247  
 ConnectionStatusSnapshotEntity (class in *nipyapi.nifi.models.connection\_status\_snapshot\_entity*), 251  
 ContentClaimContainer (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 299  
 content\_claim\_file\_size (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 299  
 content\_claim\_file\_size\_bytes (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 300  
 content\_claim\_identifier (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 300  
 content\_claim\_offset (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 300  
 content\_claim\_section (*nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO* attribute), 300  
 content\_equal (*nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO* attribute), 398  
 content\_type (*nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO* attribute), 367  
 content\_viewer\_url (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 191  
 contents (*nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO* attribute), 418  
 contents (*nipyapi.nifi.models.counter\_dto.CounterDTO* attribute), 278  
 Controller (*nipyapi.nifi.models.controller\_entity.ControllerEntity* attribute), 258  
 ControllerServiceApis (*nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO* attribute), 262  
 controller\_service\_apis (*nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO* attribute), 262  
 controller\_service\_apis (*nipyapi.nifi.models.documented\_type\_dto.DocumentedTypeDTO* attribute), 262



<i>attribute</i> ), 287	577
controller_service_apis ( <i>nipyapi.nifi.models.versioned_controller_service.VersionedControllerService</i> ), 498 <i>attribute</i> ), 498	ControllerServiceApiDTO (class in <i>nipyapi.nifi.models.versioned_controller_service.VersionedControllerService</i> ), 259
controller_service_apis ( <i>nipyapi.registry.models.versioned_controller_service.VersionedControllerService</i> ), 591 <i>attribute</i> ), 591	ControllerServiceDTO (class in <i>nipyapi.registry.models.versioned_controller_service.VersionedControllerService</i> ), 260
controller_service_bulletins ( <i>nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity</i> ), 253 <i>attribute</i> ), 253	ControllerServiceEntity (class in <i>nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity</i> ), 265
controller_service_node_results ( <i>nipyapi.nifi.models.search_results_dto.SearchResultsDTO</i> ), 445 <i>attribute</i> ), 445	ControllerServiceReferencingComponentDTO (class in <i>nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity</i> ), 267
controller_service_referencing_component ( <i>nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity</i> ), 271 <i>attribute</i> ), 271	ControllerServiceReferencingComponentEntity (class in <i>nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity</i> ), 269
controller_service_types ( <i>nipyapi.nifi.models.controller_service_types_entity.ControllerServiceTypesEntity</i> ), 272 <i>attribute</i> ), 272	ControllerServiceReferencingComponentsEntity (class in <i>nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity</i> ), 271
controller_services ( <i>nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity</i> ), 273 <i>attribute</i> ), 273	ControllerServicesApi (class in <i>nipyapi.nifi.apis.controller_services_api.ControllerServicesApi</i> ), 51
controller_services ( <i>nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO</i> ), 304 <i>attribute</i> ), 304	ControllerServicesEntity (class in <i>nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity</i> ), 273
controller_services ( <i>nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup</i> ), 519 <i>attribute</i> ), 519	ControllerServiceTypesEntity (class in <i>nipyapi.nifi.models.controller_service_types_entity.ControllerServiceTypesEntity</i> ), 272
controller_services ( <i>nipyapi.registry.models.versioned_process_group.VersionedProcessGroup</i> ), 605 <i>attribute</i> ), 605	ControllerStatusDTO (class in <i>nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity</i> ), 273
controller_status ( <i>nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity</i> ), 276 <i>attribute</i> ), 276	ControllerStatusEntity (class in <i>nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity</i> ), 276
ControllerApi (class in <i>nipyapi.nifi.apis.controller_api.ControllerApi</i> ), 43	copy_snippet () ( <i>nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi</i> method), 116
ControllerBulletinsEntity (class in <i>nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity</i> ), 253	copy_snippet_with_http_info () ( <i>nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi</i> method), 116
ControllerConfigurationDTO (class in <i>nipyapi.nifi.models.controller_configuration_dto.ControllerConfigurationDTO</i> ), 253	CopySnippetRequestEntity (class in <i>nipyapi.nifi.models.copy_snippet_request_entity.CopySnippetRequestEntity</i> ), 277
ControllerConfigurationEntity (class in <i>nipyapi.nifi.models.controller_configuration_entity.ControllerConfigurationEntity</i> ), 254	count ( <i>nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO</i> attribute), 213
ControllerDTO (class in <i>nipyapi.nifi.models.controller_dto.ControllerDTO</i> ), 255	count ( <i>nipyapi.nifi.models.batch_size.BatchSize</i> attribute), 214
ControllerEntity (class in <i>nipyapi.nifi.models.controller_entity.ControllerEntity</i> ), 258	count ( <i>nipyapi.registry.models.batch_size.BatchSize</i> attribute), 571
ControllerServiceAPI (class in <i>nipyapi.nifi.models.controller_service_api.ControllerServiceAPI</i> ), 259	counter ( <i>nipyapi.nifi.models.counter_entity.CounterEntity</i> attribute), 279
ControllerServiceAPI (class in <i>nipyapi.registry.models.controller_service_api.ControllerServiceAPI</i> ), 259	CounterDTO (class in <i>nipyapi.nifi.models.counter_dto.CounterDTO</i> ), 278
ControllerServiceAPI (class in <i>nipyapi.registry.models.controller_service_api.ControllerServiceAPI</i> ), 259	CounterEntity (class in <i>nipyapi.nifi.models.counter_entity.CounterEntity</i> ), 279

counters (*nipyapi.nifi.models.counters\_entity.CountersEntity* attribute), 280  
 counters (*nipyapi.nifi.models.counters\_snapshot\_dto.CountersSnapshotDTO* attribute), 281  
 counters\_permissions (*nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity* attribute), 284  
 CountersApi (class in *nipyapi.nifi.apis.counters\_api*), 60  
 CountersDTO (class in *nipyapi.nifi.models.counters\_dto*), 279  
 CountersEntity (class in *nipyapi.nifi.models.counters\_entity*), 280  
 CountersSnapshotDTO (class in *nipyapi.nifi.models.counters\_snapshot\_dto*), 281  
 create\_access\_policy () (in module *nipyapi.security*), 23  
 create\_access\_policy () (*nipyapi.nifi.apis.policies\_api.PoliciesApi* method), 113  
 create\_access\_policy () (*nipyapi.registry.apis.policies\_api.PoliciesApi* method), 559  
 create\_access\_policy\_with\_http\_info () (*nipyapi.nifi.apis.policies\_api.PoliciesApi* method), 113  
 create\_access\_policy\_with\_http\_info () (*nipyapi.registry.apis.policies\_api.PoliciesApi* method), 559  
 create\_access\_token () (*nipyapi.nifi.apis.access\_api.AccessApi* method), 36  
 create\_access\_token\_by\_trying\_all\_providers () (*nipyapi.registry.apis.access\_api.AccessApi* method), 537  
 create\_access\_token\_by\_trying\_all\_providers\_with\_http\_info () (*nipyapi.registry.apis.access\_api.AccessApi* method), 537  
 create\_access\_token\_from\_ticket () (*nipyapi.nifi.apis.access\_api.AccessApi* method), 36  
 create\_access\_token\_from\_ticket\_with\_http\_info () (*nipyapi.nifi.apis.access\_api.AccessApi* method), 37  
 create\_access\_token\_using\_basic\_auth\_credentials () (*nipyapi.registry.apis.access\_api.AccessApi* method), 537  
 create\_access\_token\_using\_basic\_auth\_credentials\_with\_http\_info () (*nipyapi.registry.apis.access\_api.AccessApi* method), 538  
 create\_access\_token\_using\_identity\_provider\_credentials () (*nipyapi.registry.apis.access\_api.AccessApi* method), 538  
 create\_access\_token\_using\_identity\_provider\_credentials\_with\_http\_info () (*nipyapi.registry.apis.access\_api.AccessApi* method), 538  
 create\_access\_token\_using\_kerberos\_ticket () (*nipyapi.registry.apis.access\_api.AccessApi* method), 539  
 create\_access\_token\_using\_kerberos\_ticket\_with\_http\_info () (*nipyapi.registry.apis.access\_api.AccessApi* method), 539  
 create\_access\_token\_with\_http\_info () (*nipyapi.nifi.apis.access\_api.AccessApi* method), 37  
 create\_bucket () (*nipyapi.registry.apis.buckets\_api.BucketsApi* method), 551  
 create\_bucket\_with\_http\_info () (*nipyapi.registry.apis.buckets\_api.BucketsApi* method), 551  
 create\_bulletin () (*nipyapi.nifi.apis.controller\_api.ControllerApi* method), 43  
 create\_bulletin\_with\_http\_info () (*nipyapi.nifi.apis.controller\_api.ControllerApi* method), 43  
 create\_connection () (in module *nipyapi.canvas*), 15  
 create\_connection () (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 116  
 create\_connection\_with\_http\_info () (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 117  
 create\_controller () (in module *nipyapi.canvas*), 16  
 create\_controller\_service () (*nipyapi.nifi.apis.controller\_api.ControllerApi* method), 43  
 create\_controller\_service\_with\_http\_info () (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 117  
 create\_controller\_service\_with\_http\_info () (*nipyapi.nifi.apis.controller\_api.ControllerApi* method), 44  
 create\_controller\_service\_with\_http\_info () (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 117  
 create\_drop\_request () (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi* method), 91  
 create\_drop\_request\_with\_http\_info () (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi* method), 91  
 create\_empty\_val () (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 118

`create_empty_all_connections_request_with_http_info()` (in module `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`), 118  
`create_flow()` (in module `nipyapi.versioning`), 33  
`create_flow()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` method), 543  
`create_flow_file_listing()` (`nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi` method), 91  
`create_flow_file_listing_with_http_info()` (`nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi` method), 92  
`create_flow_version()` (in module `nipyapi.versioning`), 34  
`create_flow_version()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` method), 543  
`create_flow_version_with_http_info()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` method), 544  
`create_flow_with_http_info()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` method), 544  
`create_funnel()` (in module `nipyapi.canvas`), 18  
`create_funnel()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 118  
`create_funnel_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 118  
`create_input_port()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 119  
`create_input_port_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 119  
`create_label()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 119  
`create_label_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 120  
`create_output_port()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 120  
`create_output_port_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 120  
`create_parameter_context()` (in module `nipyapi.parameters`), 20  
`create_parameter_context()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` method), 105  
`create_parameter_context_with_http_info()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` method), 105  
`create_port()` (in module `nipyapi.canvas`), 19  
`create_port_transaction()` (`nipyapi.nifi.apis.data_transfer_api.DataTransferApi` method), 63  
`create_port_transaction_with_http_info()` (`nipyapi.nifi.apis.data_transfer_api.DataTransferApi` method), 63  
`create_process_group()` (in module `nipyapi.canvas`), 12  
`create_process_group()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 120  
`create_process_group_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 121  
`create_processor()` (in module `nipyapi.canvas`), 12  
`create_processor()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 121  
`create_processor_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 121  
`create_registry_bucket()` (in module `nipyapi.versioning`), 32  
`create_registry_client()` (in module `nipyapi.versioning`), 31  
`create_registry_client()` (`nipyapi.nifi.apis.controller_api.ControllerApi` method), 44  
`create_registry_client_with_http_info()` (`nipyapi.nifi.apis.controller_api.ControllerApi` method), 44  
`create_remote_process_group()` (in module `nipyapi.canvas`), 18  
`create_remote_process_group()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 121  
`create_remote_process_group_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` method), 122  
`create_reporting_task()` (`nipyapi.nifi.apis.controller_api.ControllerApi` method), 44  
`create_reporting_task_with_http_info()` (`nipyapi.nifi.apis.controller_api.ControllerApi` method), 45  
`create_service_user()` (in module `nipyapi.security`), 21  
`create_service_user_group()` (in module `nipyapi.security`), 21  
`create_snippet()` (`nipyapi.nifi.apis.snippets_api.SnippetsApi` method), 105

*method*), 171

`create_snippet_with_http_info()`  
(*nipyapi.nifi.apis.snippets\_api.SnippetsApi*  
*method*), 171

`create_template()` (in module *nipyapi.templates*),  
27

`create_template()`  
(*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
*method*), 122

`create_template_with_http_info()`  
(*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
*method*), 122

`create_user()` (*nipyapi.nifi.apis.tenants\_api.TenantsApi*  
*method*), 175

`create_user()` (*nipyapi.registry.apis.tenants\_api.TenantsApi*  
*method*), 563

`create_user_group()`  
(*nipyapi.nifi.apis.tenants\_api.TenantsApi*  
*method*), 175

`create_user_group()`  
(*nipyapi.registry.apis.tenants\_api.TenantsApi*  
*method*), 563

`create_user_group_with_http_info()`  
(*nipyapi.nifi.apis.tenants\_api.TenantsApi*  
*method*), 176

`create_user_group_with_http_info()`  
(*nipyapi.registry.apis.tenants\_api.TenantsApi*  
*method*), 563

`create_user_with_http_info()`  
(*nipyapi.nifi.apis.tenants\_api.TenantsApi*  
*method*), 176

`create_user_with_http_info()`  
(*nipyapi.registry.apis.tenants\_api.TenantsApi*  
*method*), 563

`create_version_control_request()`  
(*nipyapi.nifi.apis.versions\_api.VersionsApi*  
*method*), 181

`create_version_control_request_with_http_info()`  
(*nipyapi.nifi.apis.versions\_api.VersionsApi*  
*method*), 181

`CreateActiveRequestEntity` (class in  
*nipyapi.nifi.models.create\_active\_request\_entity*),  
281

`created` (*nipyapi.nifi.models.bucket\_dto.BucketDTO*  
*attribute*), 216

`created_timestamp`  
(*nipyapi.nifi.models.bucket.Bucket* *attribute*),  
215

`created_timestamp`  
(*nipyapi.nifi.models.versioned\_flow.VersionedFlow*  
*attribute*), 500

`created_timestamp`  
(*nipyapi.registry.models.bucket.Bucket* *at-*  
*tribute*), 572

`created_timestamp`  
(*nipyapi.registry.models.bucket\_item.BucketItem*  
*attribute*), 573

`created_timestamp`  
(*nipyapi.registry.models.versioned\_flow.VersionedFlow*  
*attribute*), 593

`CreateTemplateRequestEntity` (class in  
*nipyapi.nifi.models.create\_template\_request\_entity*),  
282

`current` (*nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO*  
*attribute*), 289

`current_count` (*nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO*  
*attribute*), 289

`current_size` (*nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO*  
*attribute*), 289

`current_time` (*nipyapi.nifi.models.controller\_services\_entity.Controller*  
*attribute*), 273

`current_time` (*nipyapi.nifi.models.flow\_configuration\_dto.FlowConfigu*  
*attribute*), 295

`CurrentUser` (class in  
*nipyapi.registry.models.current\_user*), 577

`CurrentUserEntity` (class in  
*nipyapi.nifi.models.current\_user\_entity*),  
283

`custom_ui_url` (*nipyapi.nifi.models.controller\_service\_dto.ControllerS*  
*attribute*), 262

`custom_ui_url` (*nipyapi.nifi.models.processor\_config\_dto.ProcessorCo*  
*attribute*), 373

`custom_ui_url` (*nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskL*  
*attribute*), 436

## D

`daemon_threads` (*nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto*  
*attribute*), 462

`DataTransferApi` (class in  
*nipyapi.nifi.apis.data\_transfer\_api*), 61

`default_back_pressure_data_size_threshold`  
(*nipyapi.nifi.models.flow\_configuration\_dto.FlowConfigurationD*  
*attribute*), 295

`default_back_pressure_data_size_threshold`  
(*nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO*  
*attribute*), 351

`default_back_pressure_data_size_threshold`  
(*nipyapi.nifi.models.versioned\_process\_group.VersionedProcessC*  
*attribute*), 519

`default_back_pressure_data_size_threshold`  
(*nipyapi.registry.models.versioned\_process\_group.VersionedProc*  
*attribute*), 605

`default_back_pressure_object_threshold`  
(*nipyapi.nifi.models.flow\_configuration\_dto.FlowConfigurationD*  
*attribute*), 295

`default_back_pressure_object_threshold`  
(*nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO*  
*attribute*), 351



`default_back_pressure_object_threshold` (`nipyapi.nifi.apis.controller_api.ControllerApi`  
`(nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup`  
`attribute)`, 519 `delete_lineage()` (`nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`default_back_pressure_object_threshold` (`nipyapi.registry.models.versioned_process_group.VersionedProcessGroup`  
`attribute)`, 605 `delete_node_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`default_concurrent_tasks` (`nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO`  
`attribute)`, 373 `delete_node_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`default_flow_file_expiration` (`nipyapi.nifi.models.process_group_dto.ProcessGroupDTO`  
`attribute)`, 351 `delete_listing_request()`  
`(nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi`  
`method)`, 92  
`default_flow_file_expiration` (`nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup`  
`attribute)`, 519 `delete_listing_request_with_http_info()`  
`(nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi`  
`method)`, 92  
`default_flow_file_expiration` (`nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup`  
`attribute)`, 519 `delete_node_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`default_flow_file_expiration` (`nipyapi.registry.models.versioned_process_group.VersionedProcessGroup`  
`attribute)`, 605 `delete_node_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`default_scheduling_period` (`nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO`  
`attribute)`, 373 `delete_parameter_context()` (in module  
`nipyapi.parameters`), 20  
`default_scheduling_period` (`nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO`  
`attribute)`, 436 `delete_parameter_context()`  
`(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`  
`method)`, 106  
`default_value` (`nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO`  
`attribute)`, 390 `delete_parameter_context_with_http_info()`  
`(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`  
`method)`, 106  
`DELETE()` (`nipyapi.nifi.rest.RESTClientObject` `method`), 536 `delete_parameter_from_context()` (in mod-  
`ule nipyapi.parameters`), 21  
`DELETE()` (`nipyapi.registry.rest.RESTClientObject` `method`), 622 `delete_port()` (in module `nipyapi.canvas`), 19  
`delete_bucket()` (`nipyapi.registry.apis.buckets_api.BucketsApi` `nipyapi.canvas`), 11  
`method`), 551 `delete_process_group()` (in module  
`nipyapi.canvas`), 13  
`delete_bucket_with_http_info()` (`nipyapi.registry.apis.buckets_api.BucketsApi` `method`), 552 `delete_processor()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method)`, 141  
`delete_connection()` (in module `nipyapi.canvas`), 16 `delete_processor_with_http_info()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method)`, 141  
`delete_connection()` (`nipyapi.nifi.apis.connections_api.ConnectionsApi` `method`), 41 `delete_provenance()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`delete_connection_with_http_info()` (`nipyapi.nifi.apis.connections_api.ConnectionsApi` `method`), 41 `delete_provenance_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method)`, 149  
`delete_controller()` (in module `nipyapi.canvas`), 16 `delete_registry_bucket()` (in module  
`nipyapi.versioning`), 32  
`delete_flow()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` `method`), 544 `delete_registry_client()` (in module  
`nipyapi.versioning`), 31  
`delete_flow_with_http_info()` (`nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi` `method`), 545 `delete_registry_client()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method)`, 46  
`delete_funnel()` (in module `nipyapi.canvas`), 18 `delete_registry_client_with_http_info()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method)`, 45  
`delete_history()` (`nipyapi.nifi.apis.controller_api.ControllerApi` `method`), 45  
`delete_history_with_http_info()` (`nipyapi.nifi.apis.controller_api.ControllerApi` `method`), 45

method), 46

delete\_remote\_process\_group() (in module `nipyapi.canvas`), 18

delete\_replace\_process\_group\_request() (in module `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`), 123

delete\_replace\_process\_group\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`), 123

delete\_revert\_request() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 181

delete\_revert\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 182

delete\_snippet() (in module `nipyapi.nifi.apis.snippets_api.SnippetsApi`), 171

delete\_snippet\_with\_http\_info() (in module `nipyapi.nifi.apis.snippets_api.SnippetsApi`), 172

delete\_template() (in module `nipyapi.templates`), 27

delete\_update\_request() (in module `nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`), 107

delete\_update\_request() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 182

delete\_update\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`), 107

delete\_update\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 183

delete\_validation\_request() (in module `nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`), 52

delete\_validation\_request() (in module `nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`), 107

delete\_validation\_request() (in module `nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`), 163

delete\_validation\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`), 53

delete\_validation\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi`), 108

delete\_validation\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`), 163

delete\_variable\_registry\_update\_request() (in module `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`), 123

delete\_variable\_registry\_update\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`), 124

delete\_verification\_request() (in module `nipyapi.nifi.apis.processors_api.ProcessorsApi`), 142

delete\_verification\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.processors_api.ProcessorsApi`), 142

delete\_version\_control\_request() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 183

delete\_version\_control\_request\_with\_http\_info() (in module `nipyapi.nifi.apis.versions_api.VersionsApi`), 183

dependencies (in module `nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO`), 390

deploy\_flow\_version() (in module `nipyapi.versioning`), 35

deploy\_template() (in module `nipyapi.templates`), 26

deprecated (in module `nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO`), 262

deprecated (in module `nipyapi.nifi.models.processor_dto.ProcessorDTO`), 376

deprecated (in module `nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO`), 436

deprecation\_reason (in module `nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO`), 288

description (in module `nipyapi.nifi.models.allowable_value_dto.AllowableValueDTO`), 209

description (in module `nipyapi.nifi.models.bucket.Bucket`), 215

description (in module `nipyapi.nifi.models.bucket_dto.BucketDTO`), 216

description (in module `nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity`), 282

description (in module `nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO`), 288

description (in module `nipyapi.nifi.models.processor_dto.ProcessorDTO`), 376

description (in module `nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO`), 391

description (in module `nipyapi.nifi.models.registry_dto.RegistryDTO`), 414

description (in module `nipyapi.nifi.models.relationship_dto.RelationshipDTO`), 415

description (in module `nipyapi.nifi.models.status_descriptor_dto.StatusDescriptorDTO`), 453

description (in module `nipyapi.nifi.models.template_dto.TemplateDTO`), 465

description (in module `nipyapi.nifi.models.variable_registry_update_step_dto.VariableRegistryUpdateStepDTO`), 465

attribute), 487  
 description (nipyapi.nifi.models.versioned\_flow.VersionedFlow attribute), 500  
 description (nipyapi.nifi.models.versioned\_flow\_dto.VersionedFlowDTO attribute), 503  
 description (nipyapi.registry.models.bucket.Bucket attribute), 572  
 description (nipyapi.registry.models.bucket\_item.BucketItem attribute), 574  
 description (nipyapi.registry.models.versioned\_flow.VersionedFlow attribute), 593  
 descriptors (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO attribute), 262  
 descriptors (nipyapi.nifi.models.controller\_service\_reference\_dto.ControllerServiceReferenceDTO attribute), 268  
 descriptors (nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO attribute), 373  
 descriptors (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 436  
 deserialize() (nipyapi.nifi.api\_client.ApiClient method), 534  
 deserialize() (nipyapi.registry.api\_client.ApiClient method), 620  
 deserialize\_model() (nipyapi.nifi.api\_client.ApiClient method), 534  
 deserialize\_model() (nipyapi.registry.api\_client.ApiClient method), 620  
 destination (nipyapi.nifi.models.connection\_dto.ConnectionDTO attribute), 240  
 destination (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 495  
 destination (nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 587  
 destination\_group\_id (nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 242  
 destination\_id (nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 243  
 destination\_id (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO attribute), 245  
 destination\_id (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO attribute), 249  
 destination\_name (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO attribute), 245  
 destination\_name (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO attribute), 249  
 destination\_running (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO attribute), 320  
 destination\_type (nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 243  
 details (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 282  
 attribute), 398  
 description (nipyapi.nifi.models.difference\_dto.DifferenceDTO attribute), 286  
 description (nipyapi.nifi.models.difference\_dto.DifferenceDTO attribute), 286  
 description (nipyapi.nifi.models.difference\_dto.DifferenceDTO (class in nipyapi.nifi.models.difference\_dto), 286  
 description (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO attribute), 228  
 description (nipyapi.nifi.models.label\_entity.LabelEntity attribute), 314  
 descriptors (nipyapi.nifi.models.dimensions\_dto.DimensionsDTO (class in nipyapi.nifi.models.dimensions\_dto), 286  
 descriptors (nipyapi.nifi.models.controller\_service\_reference\_dto.ControllerServiceReferenceDTO attribute), 256  
 descriptors (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO attribute), 274  
 descriptors (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 351  
 disabled\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 357  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity attribute), 196  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.access\_policy\_summary\_entity.AccessPolicySummaryEntity attribute), 199  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.activate\_controller\_services\_entity.ActivateControllerServicesEntity attribute), 205  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity attribute), 208  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.component\_reference\_entity.ComponentReferenceEntity attribute), 232  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 243  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.controller\_configuration\_entity.ControllerConfigurationEntity attribute), 255  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity attribute), 266  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.status\_snapshot\_entity.StatusSnapshotEntity attribute), 270  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.copy\_snippet\_request\_entity.CopySnippetRequestEntity attribute), 277  
 disconnected\_node\_acknowledged (nipyapi.nifi.models.create\_active\_request\_entity.CreateActiveRequestEntity attribute), 282

disconnected_node_acknowledged ( <i>nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity</i> attribute), 282	disconnected_node_acknowledged ( <i>nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity</i> attribute), 478
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.funnel_entity.FunnelEntity</i> attribute), 307	disconnected_node_acknowledged ( <i>nipyapi.nifi.models.variable_registry_entity.VariableRegistryEntity</i> attribute), 483
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity</i> attribute), 311	disconnected_node_acknowledged ( <i>nipyapi.nifi.models.control_component_mapping_entity.ControlComponentMappingEntity</i> attribute), 488
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.label_entity.LabelEntity</i> attribute), 315	disconnected_node_acknowledged ( <i>nipyapi.nifi.models.version_control_information_entity.VersionControlInformationEntity</i> attribute), 491
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.port_entity.PortEntity</i> attribute), 340	disconnected_node_acknowledged ( <i>nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity</i> attribute), 506
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.process_group_entity.ProcessGroupEntity</i> attribute), 357	display_name ( <i>nipyapi.nifi.models.allowable_value_dto.AllowableValueDto</i> attribute), 209
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.processor_entity.ProcessorEntity</i> attribute), 380	display_name ( <i>nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDto</i> attribute), 391
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.processor_entity.ProcessorEntity</i> attribute), 380	display_name ( <i>nipyapi.nifi.models.versioned_property_descriptor.VersionedPropertyDescriptor</i> attribute), 526
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.registry_client_entity.RegistryClientEntity</i> attribute), 412	display_name ( <i>nipyapi.registry.models.versioned_property_descriptor.VersionedPropertyDescriptor</i> attribute), 612
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity</i> attribute), 422	DockerContainer (class in <i>nipyapi.utils</i> ), 30
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity</i> attribute), 422	DocumentedTypeDTO (class in <i>nipyapi.registry.models.documented_type_dto</i> ), 287
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.remote_process_group_port_entity.RemoteProcessGroupPortEntity</i> attribute), 427	download_flow_file_content () ( <i>nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi</i> method), 93
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.reporting_task_entity.ReportingTaskEntity</i> attribute), 439	download_flow_file_content_with_http_info () ( <i>nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi</i> method), 93
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.schedule_components_entity.ScheduleComponentsEntity</i> attribute), 444	drop_request ( <i>nipyapi.nifi.models.drop_request_entity.DropRequestEntity</i> attribute), 289
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.snippet_entity.SnippetEntity</i> attribute), 449	dropped ( <i>nipyapi.nifi.models.drop_request_dto.DropRequestDTO</i> attribute), 289
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.start_version_control_request_entity.StartVersionControlRequestEntity</i> attribute), 450	dropped_count ( <i>nipyapi.nifi.models.drop_request_dto.DropRequestDTO</i> attribute), 289
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.start_version_control_request_entity.StartVersionControlRequestEntity</i> attribute), 450	dropped_size ( <i>nipyapi.nifi.models.drop_request_dto.DropRequestDTO</i> attribute), 289
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.template_entity.TemplateEntity</i> attribute), 467	DropRequestDTO (class in <i>nipyapi.nifi.models.drop_request_dto</i> ), 289
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.template_entity.TemplateEntity</i> attribute), 467	DropRequestEntity (class in <i>nipyapi.nifi.models.drop_request_entity</i> ), 291
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.tenant_entity.TenantEntity</i> attribute), 470	dump () (in module <i>nipyapi.utils</i> ), 28
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.update_controller_service_reference_request_entity.UpdateControllerServiceReferenceRequestEntity</i> attribute), 473	duration ( <i>nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO</i> attribute), 213
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.update_controller_service_reference_request_entity.UpdateControllerServiceReferenceRequestEntity</i> attribute), 473	duration ( <i>nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO</i> attribute), 213
disconnected_node_acknowledged ( <i>nipyapi.nifi.models.user_entity.UserEntity</i> attribute), 476	dynamic ( <i>nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDto</i> attribute), 571
	dynamic ( <i>nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDto</i> attribute), 571



attribute), 391

**E**

encoding\_version(nipyapi.nifi.models.instantiate\_template\_request\_entity.InstantiateTemplateRequestEntity attribute), 312

encoding\_version(nipyapi.nifi.models.template\_dto.TemplateDTO attribute), 465

end\_date(nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO attribute), 407

enforce\_min\_ver() (in module nipyapi.utils), 30

errors(nipyapi.nifi.models.lineage\_results\_dto.LineageResultsDTO attribute), 319

errors(nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO attribute), 409

event\_duration(nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 398

event\_id(nipyapi.nifi.models.lineage\_request\_dto.LineageRequestDTO attribute), 318

event\_id(nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 398

event\_id(nipyapi.nifi.models.submit\_replay\_request\_entity.SubmitReplayRequestEntity attribute), 458

event\_time(nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 398

event\_type(nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 398

event\_type(nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO attribute), 404

events (nipyapi.nifi.models.node\_dto.NodeDTO attribute), 325

exception\_handler() (in module nipyapi.utils), 30

execution\_node(nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO attribute), 373

execution\_node(nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 385

execution\_node(nipyapi.nifi.models.versioned\_processor\_versioned\_processor attribute), 523

execution\_node(nipyapi.registry.models.versioned\_processor\_versioned\_processor attribute), 609

execution\_node\_restricted (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 376

exists(nipyapi.nifi.models.connectable\_dto.ConnectableDTO attribute), 237

exists(nipyapi.nifi.models.remote\_process\_group\_port\_dto.RemoteProcessGroupPortDTO attribute), 425

expiration(nipyapi.nifi.models.lineage\_dto.LineageDTO attribute), 316

expiration(nipyapi.nifi.models.provenance\_dto.ProvenanceDTO attribute), 393

explicit\_restrictions (nipyapi.nifi.models.documented\_type\_dto.DocumentedTypeDTO attribute), 288

export\_flow\_version() (in module nipyapi.versioning), 34

export\_flow\_version() (nipyapi.nifi.apis.versions\_api.VersionsApi method), 184

export\_flow\_version\_with\_http\_info() (nipyapi.nifi.apis.versions\_api.VersionsApi method), 184

export\_process\_group() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 124

export\_process\_group\_with\_http\_info() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 124

export\_template() (in module nipyapi.templates), 27

export\_template() (nipyapi.nifi.apis.templates\_api.TemplatesApi method), 174

export\_template\_with\_http\_info() (nipyapi.nifi.apis.templates\_api.TemplatesApi method), 174

export\_versioned\_flow() (nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi method), 545

export\_versioned\_flow\_with\_http\_info() (nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi method), 545

expression\_language\_scope (nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDTO attribute), 391

extend\_input\_port\_transaction\_ttl() (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi method), 63

extend\_input\_port\_transaction\_ttl\_with\_http\_info() (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi method), 63

extend\_output\_port\_transaction\_ttl() (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi method), 64

extend\_output\_port\_transaction\_ttl\_with\_http\_info() (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi method), 64

extension\_missing (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 262

extension\_missing (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 377

extension\_missing (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 436

external\_controller\_services (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot attribute), 440

attribute), 505  
 external\_controller\_services  
 (nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 596  
 attribute), 596  
**F**  
 failure\_reason (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO), 290  
 failure\_reason (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO), 320  
 failure\_reason (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO), 485  
 failure\_reason (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO), 487  
 failure\_reason (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO), 509  
 field (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO), 410  
 field (nipyapi.nifi.models.status\_descriptor\_dto.StatusDescriptorDTO), 453  
 field\_descriptors  
 (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO), 454  
 Fields (class in nipyapi.registry.models.fields), 578  
 fields (nipyapi.registry.models.fields.Fields), 579  
 file\_size (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO), 398  
 file\_size\_bytes (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO), 398  
 filename (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO), 300  
 filename (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO), 302  
 filter\_obj() (in module nipyapi.utils), 29  
 finished (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO), 290  
 finished (nipyapi.nifi.models.lineage\_dto.LineageDTO), 316  
 finished (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO), 321  
 finished (nipyapi.nifi.models.provenance\_dto.ProvenanceDTO), 393  
 flow (nipyapi.nifi.models.flow\_entity.FlowEntity), 298  
 flow (nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO), 361  
 flow (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 505  
 flow (nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 596  
 flow\_configuration  
 (nipyapi.nifi.models.flow\_configuration\_entity.FlowConfigurationEntity), 296  
 flow\_contents (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 505  
 attribute), 505  
 flow\_description (nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO), 489  
 attribute), 489  
 flow\_encoding\_version  
 (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 505  
 flow\_encoding\_version  
 (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot), 596  
 flow\_file\_entity (nipyapi.nifi.models.flow\_file\_entity.FlowFileEntity), 301  
 attribute), 301  
 flow\_file\_availability (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO), 509  
 (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO), 509  
 field (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO), 410  
 flow\_file\_concurrency  
 (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup), 519  
 attribute), 519  
 flow\_file\_concurrency  
 (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup), 605  
 attribute), 605  
 flow\_file\_count (nipyapi.nifi.models.peer\_dto.PeerDTO), 335  
 attribute), 335  
 flow\_file\_expiration  
 (nipyapi.nifi.models.connection\_dto.ConnectionDTO), 240  
 attribute), 240  
 flow\_file\_expiration  
 (nipyapi.nifi.models.versioned\_connection.VersionedConnectionDTO), 495  
 attribute), 495  
 flow\_file\_expiration  
 (nipyapi.registry.models.versioned\_connection.VersionedConnectionDTO), 587  
 attribute), 587  
 flow\_file\_outbound\_policy  
 (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup), 519  
 attribute), 519  
 flow\_file\_outbound\_policy  
 (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup), 605  
 attribute), 605  
 flow\_file\_repository\_storage\_usage  
 (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO), 462  
 attribute), 462  
 flow\_file\_sent (nipyapi.nifi.models.transaction\_result\_entity.TransactionResultEntity), 472  
 attribute), 472  
 flow\_file\_summaries  
 (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO), 321  
 attribute), 321  
 flow\_file\_uuid (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO), 398  
 attribute), 398  
 flow\_file\_uuid (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO), 402  
 attribute), 402  
 flow\_files\_data (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO), 404  
 attribute), 404

`flow_files_in` (`nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO`), 66  
 attribute), 249 `FlowBreadcrumbDTO` (class in `nipyapi.nifi.models.flow_breadcrumb_entity`), 292  
`flow_files_in` (`nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO`), 345  
 attribute), 345  
`flow_files_in` (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367 `nipyapi.nifi.models.flow_breadcrumb_entity`, 292  
`flow_files_in` (`nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO`), 385  
 attribute), 385 `FlowComparisonEntity` (class in `nipyapi.nifi.models.flow_comparison_entity`), 294  
`flow_files_out` (`nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO`), 249  
 attribute), 249  
`flow_files_out` (`nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO`), 345  
 attribute), 345 `nipyapi.nifi.models.flow_configuration_dto`, 294  
`flow_files_out` (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367 `FlowConfigurationEntity` (class in `nipyapi.nifi.models.flow_configuration_entity`), 296  
`flow_files_out` (`nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO`), 385  
 attribute), 385  
`flow_files_queued` `FlowDTO` (class in `nipyapi.nifi.models.flow_dto`), 297  
 (`nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO`), 298  
 attribute), 249  
`flow_files_queued` `flowfile_concurrency`  
 (`nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO`), 274  
 attribute), 274 `nipyapi.nifi.models.process_group_dto.ProcessGroupDTO`  
 attribute), 351  
`flow_files_queued` `flowfile_outbound_policy`  
 (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367  
`flow_files_received` `FlowFileDTO` (class in `nipyapi.nifi.models.flow_file_dto`), 299  
 (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367 `FlowFileEntity` (class in `nipyapi.nifi.models.flow_file_entity`), 301  
`flow_files_received` `nipyapi.nifi.models.flow_file_entity`, 301  
 (`nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO`), 431  
 attribute), 431 `nipyapi.nifi.apis.flowfile_queues_api`, 91  
`flow_files_sent` (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367 `nipyapi.nifi.models.flow_file_summary_dto`, 303  
`flow_files_sent` (`nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO`), 431  
 attribute), 431 `FlowsApi` (class in `nipyapi.registry.apis.flows_api`), 554  
`flow_files_transferred` 554  
 (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO`), 367  
 attribute), 367 `nipyapi.nifi.models.flow_snippet_dto`, 304  
`flow_id` (`nipyapi.nifi.models.version_control_information_dto.VersionControlInformationDTO`), 489  
 attribute), 489  
`flow_id` (`nipyapi.nifi.models.versioned_flow_coordinates.VersionedFlowCoordinates`), 502  
 attribute), 502 `nipyapi.nifi.models.status_descriptor_dto.StatusDescriptorDTO`  
 attribute), 453  
`flow_id` (`nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO`), 503  
 attribute), 503 `nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO`  
 attribute), 462  
`flow_id` (`nipyapi.registry.models.versioned_flow_coordinates.VersionedFlowCoordinates`), 594  
 attribute), 594  
`flow_identifier` (`nipyapi.registry.models.versioned_flow_coordinates.VersionedFlowCoordinates`), 597  
 attribute), 597  
`flow_name` (`nipyapi.nifi.models.version_control_information_dto.VersionControlInformationDTO`), 489  
 attribute), 489 `nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO`  
 attribute), 462  
`flow_name` (`nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO`), 503  
 attribute), 503 `free_space` (`nipyapi.nifi.models.storage_usage_dto.StorageUsageDTO`), 476  
`flow_refreshed` (`nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO`), 418  
 attribute), 418 `free_space_bytes` (`nipyapi.nifi.models.storage_usage_dto.StorageUsageDTO`), 476

attribute), 456  
 fs\_read() (in module nipyapi.utils), 28  
 fs\_write() (in module nipyapi.utils), 28  
 funnel\_results (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO attribute), 445  
 FunnelApi (class in nipyapi.nifi.apis.funnel\_api), 96  
 FunnelDTO (class in nipyapi.nifi.models.funnel\_dto), 305  
 FunnelEntity (class in nipyapi.nifi.models.funnel\_entity), 306  
 funnels (nipyapi.nifi.models.flow\_dto.FlowDTO attribute), 297  
 funnels (nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO attribute), 304  
 funnels (nipyapi.nifi.models.funnels\_entity.FunnelsEntity attribute), 308  
 funnels (nipyapi.nifi.models.snippet\_dto.SnippetDTO attribute), 448  
 funnels (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520  
 funnels (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606  
 FunnelsEntity (class in nipyapi.nifi.models.funnels\_entity), 308  
**G**  
 garbage\_collection (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO attribute), 462  
 GarbageCollectionDTO (class in nipyapi.nifi.models.garbage\_collection\_dto), 308  
 generate\_client\_id() (nipyapi.nifi.apis.flow\_api.FlowApi method), 66  
 generate\_client\_id\_with\_http\_info() (nipyapi.nifi.apis.flow\_api.FlowApi method), 67  
 generated (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity attribute), 196  
 generated (nipyapi.nifi.models.bulletin\_board\_dto.BulletinBoardDTO attribute), 218  
 generated (nipyapi.nifi.models.cluster\_dto.ClusterDTO attribute), 224  
 generated (nipyapi.nifi.models.counters\_snapshot\_dto.CountersSnapshotDTO attribute), 281  
 generated (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO attribute), 409  
 generated (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO attribute), 454  
 generated (nipyapi.nifi.models.templates\_entity.TemplatesEntity attribute), 468  
 generated (nipyapi.nifi.models.users\_entity.UsersEntity attribute), 480  
 GET () (nipyapi.nifi.rest.RESTClientObject method), 536  
 GET () (nipyapi.registry.rest.RESTClientObject method), 622  
 get\_search\_results() (nipyapi.nifi.apis.flow\_api.FlowApi method), 67  
 get\_about\_info\_with\_http\_info() (nipyapi.nifi.apis.flow\_api.FlowApi method), 67  
 get\_access\_policies() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 559  
 get\_access\_policies\_with\_http\_info() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 559  
 get\_access\_policy() (nipyapi.nifi.apis.policies\_api.PoliciesApi method), 113  
 get\_access\_policy() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 560  
 get\_access\_policy\_for\_resource() (in module nipyapi.security), 23  
 get\_access\_policy\_for\_resource() (nipyapi.nifi.apis.policies\_api.PoliciesApi method), 113  
 get\_access\_policy\_for\_resource() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 560  
 get\_access\_policy\_for\_resource\_with\_http\_info() (nipyapi.nifi.apis.policies\_api.PoliciesApi method), 114  
 get\_access\_policy\_for\_resource\_with\_http\_info() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 560  
 get\_access\_policy\_with\_http\_info() (nipyapi.nifi.apis.policies\_api.PoliciesApi method), 114  
 get\_access\_policy\_with\_http\_info() (nipyapi.registry.apis.policies\_api.PoliciesApi method), 560  
 get\_access\_status() (nipyapi.nifi.apis.access\_api.AccessApi method), 37  
 get\_access\_status() (nipyapi.registry.apis.access\_api.AccessApi method), 539  
 get\_access\_status\_with\_http\_info() (nipyapi.nifi.apis.access\_api.AccessApi method), 37  
 get\_access\_status\_with\_http\_info() (nipyapi.registry.apis.access\_api.AccessApi method), 539  
 get\_access\_token\_expiration() (nipyapi.nifi.apis.access\_api.AccessApi



*method*), 38  
 get\_access\_token\_expiration\_with\_http\_info() (in module *nipyapi.access\_api.AccessApi* *method*), 38  
 get\_action() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 67  
 get\_action\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 68  
 get\_available\_bucket\_fields() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 552  
 get\_available\_bucket\_fields\_with\_http\_info() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 552  
 get\_available\_bucket\_item\_fields() (*nipyapi.registry.apis.items\_api.ItemsApi* *method*), 557  
 get\_available\_bucket\_item\_fields\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi* *method*), 557  
 get\_available\_flow\_fields() (*nipyapi.registry.apis.flows\_api.FlowsApi* *method*), 554  
 get\_available\_flow\_fields\_with\_http\_info() (*nipyapi.registry.apis.flows\_api.FlowsApi* *method*), 554  
 get\_banners() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 68  
 get\_banners\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 68  
 get\_bucket() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 552  
 get\_bucket\_with\_http\_info() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 553  
 get\_buckets() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 68  
 get\_buckets() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 553  
 get\_buckets\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 68  
 get\_buckets\_with\_http\_info() (*nipyapi.registry.apis.buckets\_api.BucketsApi* *method*), 553  
 get\_bulletin\_board() (in module *nipyapi.canvas*), 15  
 get\_bulletin\_board() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 69  
 get\_bulletin\_board\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 69  
 get\_bulletins() (in module *nipyapi.canvas*), 15  
 get\_bulletins() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 70  
 get\_bulletins\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 70  
 get\_cluster() (in module *nipyapi.system*), 25  
 get\_cluster() (*nipyapi.nifi.apis.controller\_api.ControllerApi* *method*), 47  
 get\_cluster\_summary() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 70  
 get\_cluster\_summary\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 70  
 get\_cluster\_with\_http\_info() (*nipyapi.nifi.apis.controller\_api.ControllerApi* *method*), 47  
 get\_component\_connections() (in module *nipyapi.canvas*), 16  
 get\_component\_history() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 70  
 get\_component\_history\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 71  
 get\_connection() (*nipyapi.nifi.apis.connections\_api.ConnectionsApi* *method*), 42  
 get\_connection\_statistics() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 71  
 get\_connection\_statistics\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 71  
 get\_connection\_status() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 72  
 get\_connection\_status\_history() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 72  
 get\_connection\_status\_history\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 72  
 get\_connection\_status\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* *method*), 72  
 get\_connection\_with\_http\_info() (*nipyapi.nifi.apis.connections\_api.ConnectionsApi* *method*), 42  
 get\_connections() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* *method*), 125  
 get\_connections\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* *method*), 125

(*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* 76  
 method), 125  
 get\_container() (*nipyapi.utils.DockerContainer*  
 method), 30  
 get\_controller() (*in module nipyapi.canvas*), 17  
 get\_controller\_config() (*nipyapi.nifi.apis.controller\_api.ControllerApi*  
 method), 47  
 get\_controller\_config\_with\_http\_info() (*nipyapi.nifi.apis.controller\_api.ControllerApi*  
 method), 47  
 get\_controller\_service() (*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi*  
 method), 53  
 get\_controller\_service\_references() (*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi*  
 method), 53  
 get\_controller\_service\_references\_with\_http\_info() (*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi*  
 method), 54  
 get\_controller\_service\_types() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 73  
 get\_controller\_service\_types\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 73  
 get\_controller\_service\_with\_http\_info() (*nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi*  
 method), 54  
 get\_controller\_services\_from\_controller() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 74  
 get\_controller\_services\_from\_controller\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 74  
 get\_controller\_services\_from\_group() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 74  
 get\_controller\_services\_from\_group\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 75  
 get\_controller\_status() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 75  
 get\_controller\_status\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 76  
 get\_counters() (*nipyapi.nifi.apis.counters\_api.CountersApi* method), 60  
 get\_counters\_with\_http\_info() (*nipyapi.nifi.apis.counters\_api.CountersApi* method), 60  
 get\_current\_user() (*nipyapi.nifi.apis.flow\_api.FlowApi* method),  
 get\_current\_user\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 76  
 get\_drop\_all\_flowfiles\_request() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 125  
 get\_drop\_all\_flowfiles\_request\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 125  
 get\_drop\_request() (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi* method), 93  
 get\_drop\_request\_with\_http\_info() (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi* method), 93  
 get\_flow() (*in module nipyapi.canvas*), 10  
 get\_flow() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 76  
 get\_flow() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 546  
 get\_flow\_config() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 76  
 get\_flow\_config\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 77  
 get\_flow\_diff() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 546  
 get\_flow\_diff\_with\_http\_info() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 546  
 get\_flow\_file\_with\_http\_info() (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi* method), 94  
 get\_flow\_in\_bucket() (*in module nipyapi.versioning*), 33  
 get\_flow\_metrics() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 77  
 get\_flow\_metrics\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 77  
 get\_flow\_version() (*in module nipyapi.versioning*), 34  
 get\_flow\_version() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 546  
 get\_flow\_version\_with\_http\_info() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 547  
 get\_flow\_versions()

(*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 98  
 method), 547  
 get\_flow\_versions\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
 (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 126  
 method), 547  
 get\_flow\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
 (*nipyapi.nifi.apis.flow\_api.FlowApi* method), method), 126  
 78  
 get\_flow\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi*  
 method), 558  
 (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 547  
 method), 547  
 get\_flows() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 78  
 method), 78  
 get\_flows() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi* method), 548  
 method), 548  
 get\_flows\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 78  
 78  
 get\_flows\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi*  
 method), 558  
 method), 558  
 get\_flows\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi*  
 method), 558  
 method), 558  
 get\_funnel() (in module *nipyapi.canvas*), 13  
 get\_funnel() (*nipyapi.nifi.apis.funnel\_api.FunnelApi* method), 101  
 method), 96  
 get\_funnel\_with\_http\_info() (*nipyapi.nifi.apis.funnel\_api.FunnelApi*  
 method), 96  
 method), 96  
 get\_funnels() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 127  
 method), 126  
 get\_funnels\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
 method), 126  
 method), 126  
 get\_identity\_provider\_usage\_instructions() (*nipyapi.registry.apis.access\_api.AccessApi*  
 method), 540  
 method), 540  
 get\_identity\_provider\_usage\_instructions\_with\_http\_info() (*nipyapi.registry.apis.access\_api.AccessApi*  
 method), 540  
 method), 540  
 get\_input\_content() (*nipyapi.nifi.apis.provenance\_events\_api.ProvenanceEventsApi*  
 method), 153  
 method), 153  
 get\_input\_content\_with\_http\_info() (*nipyapi.nifi.apis.provenance\_events\_api.ProvenanceEventsApi*  
 method), 153  
 method), 153  
 get\_input\_port() (*nipyapi.nifi.apis.input\_ports\_api.InputPortsApi* method), 98  
 method), 98  
 get\_input\_port\_status() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 79  
 79  
 get\_input\_port\_status\_with\_http\_info() (*nipyapi.nifi.apis.flow\_api.FlowApi* method), 79  
 79  
 get\_input\_port\_with\_http\_info() (*nipyapi.nifi.apis.input\_ports\_api.InputPortsApi* method), 95  
 method), 95  
 get\_input\_ports() (*nipyapi.registry.apis.items\_api.ItemsApi* method), 558  
 method), 558  
 get\_items() (*nipyapi.registry.apis.items\_api.ItemsApi* method), 558  
 method), 558  
 get\_items\_in\_bucket() (*nipyapi.registry.apis.items\_api.ItemsApi* method), 558  
 method), 558  
 get\_items\_in\_bucket\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi*  
 method), 558  
 method), 558  
 get\_items\_with\_http\_info() (*nipyapi.registry.apis.items\_api.ItemsApi* method), 558  
 method), 558  
 get\_label() (*nipyapi.nifi.apis.labels\_api.LabelsApi* method), 100  
 method), 100  
 get\_label\_with\_http\_info() (*nipyapi.nifi.apis.labels\_api.LabelsApi* method), 101  
 method), 101  
 get\_labels() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi* method), 127  
 method), 127  
 get\_labels\_with\_http\_info() (*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
 method), 127  
 method), 127  
 get\_latest\_flow\_ver() (in module *nipyapi.versioning*), 33  
 get\_latest\_flow\_version() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi*  
 method), 548  
 method), 548  
 get\_latest\_flow\_version\_metadata() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi*  
 method), 549  
 method), 549  
 get\_latest\_flow\_version\_metadata\_with\_http\_info() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi*  
 method), 549  
 method), 549  
 get\_latest\_flow\_version\_with\_http\_info() (*nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi*  
 method), 549  
 method), 549  
 get\_latest\_flow\_version\_with\_http\_info() (*nipyapi.nifi.apis.provenance\_api.ProvenanceApi*  
 method), 150  
 method), 150  
 get\_latest\_flow\_version\_with\_http\_info() (*nipyapi.nifi.apis.provenance\_api.ProvenanceApi*  
 method), 150  
 method), 150  
 get\_listing\_request() (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi*  
 method), 94  
 method), 94  
 get\_listing\_request\_with\_http\_info() (*nipyapi.nifi.apis.flowfile\_queues\_api.FlowfileQueuesApi*  
 method), 95  
 method), 95  
 get\_local\_modifications()

```
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 127
get_local_modifications_with_http_info()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 127
get_login_config()
(nipyapi.nifi.apis.access_api.AccessApi
method), 38
get_login_config_with_http_info()
(nipyapi.nifi.apis.access_api.AccessApi
method), 38
get_nifi_version_info() (in module
nipyapi.system), 26
get_node() (in module nipyapi.system), 26
get_node() (nipyapi.nifi.apis.controller_api.ControllerApi
method), 47
get_node_status_history()
(nipyapi.nifi.apis.controller_api.ControllerApi
method), 48
get_node_status_history_with_http_info()
(nipyapi.nifi.apis.controller_api.ControllerApi
method), 48
get_node_with_http_info()
(nipyapi.nifi.apis.controller_api.ControllerApi
method), 48
get_output_content()
(nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi
method), 153
get_output_content_with_http_info()
(nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi
method), 154
get_output_port()
(nipyapi.nifi.apis.output_ports_api.OutputPortsApi
method), 103
get_output_port_status()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 79
get_output_port_status_with_http_info()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 80
get_output_port_with_http_info()
(nipyapi.nifi.apis.output_ports_api.OutputPortsApi
method), 103
get_output_ports()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 128
get_output_ports_with_http_info()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 128
get_parameter_context() (in module
nipyapi.parameters), 20
get_parameter_context()
(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi
method), 108
get_parameter_context_update()
(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi
method), 108
get_parameter_context_update_with_http_info()
(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi
method), 109
get_parameter_context_with_http_info()
(nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi
method), 109
get_parameter_contexts()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 80
get_parameter_contexts_with_http_info()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 80
get_peers() (nipyapi.nifi.apis.site_to_site_api.SiteToSiteApi
method), 170
get_peers_with_http_info()
(nipyapi.nifi.apis.site_to_site_api.SiteToSiteApi
method), 170
get_pg_parents_ids() (in module
nipyapi.canvas), 19
get_prioritizers()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 80
get_prioritizers_with_http_info()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 81
get_process_group() (in module nipyapi.canvas),
get_process_group()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 128
get_process_group_status() (in module
nipyapi.canvas), 10
get_process_group_status()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 81
get_process_group_status_history()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 81
get_process_group_status_history_with_http_info()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 81
get_process_group_status_with_http_info()
(nipyapi.nifi.apis.flow_api.FlowApi
method), 82
get_process_group_with_http_info()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 128
get_process_groups()
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 129
get_process_groups_with_http_info()
```

`(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
`method), 129`  
`get_processor() (in module nipyapi.canvas), 13`  
`get_processor() (nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 142`  
`get_processor_diagnostics()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 143`  
`get_processor_diagnostics_with_http_info()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 143`  
`get_processor_run_status_details()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 143`  
`get_processor_run_status_details_with_http_info()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 143`  
`get_processor_status()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 82`  
`get_processor_status_history()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 82`  
`get_processor_status_history_with_http_info()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 83`  
`get_processor_status_with_http_info()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 83`  
`get_processor_type() (in module`  
`nipyapi.canvas), 12`  
`get_processor_types()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 83`  
`get_processor_types_with_http_info()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 84`  
`get_processor_with_http_info()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 144`  
`get_processors() (nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
`method), 129`  
`get_processors_with_http_info()`  
`(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
`method), 129`  
`get_property_descriptor()`  
`(nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`  
`method), 54`  
`get_property_descriptor()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 144`  
`get_property_descriptor()`  
`(nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`  
`method), 164`  
`get_property_descriptor_with_http_info()`  
`(nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`  
`method), 54`  
`get_property_descriptor_with_http_info()`  
`(nipyapi.nifi.apis.processors_api.ProcessorsApi`  
`method), 144`  
`get_property_descriptor_with_http_info()`  
`(nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`  
`method), 164`  
`get_provenance()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method), 150`  
`get_provenance_event()`  
`(nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi`  
`method), 154`  
`get_provenance_event_with_http_info()`  
`(nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi`  
`method), 154`  
`get_provenance_with_http_info()`  
`(nipyapi.nifi.apis.provenance_api.ProvenanceApi`  
`method), 151`  
`get_registries()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 84`  
`get_registries_with_http_info()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 84`  
`get_registry_bucket() (in module`  
`nipyapi.versioning), 32`  
`get_registry_client() (in module`  
`nipyapi.versioning), 31`  
`get_registry_client()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method), 48`  
`get_registry_client_with_http_info()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method), 49`  
`get_registry_clients()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method), 49`  
`get_registry_clients_with_http_info()`  
`(nipyapi.nifi.apis.controller_api.ControllerApi`  
`method), 49`  
`get_remote_process_group() (in module`  
`nipyapi.canvas), 18`  
`get_remote_process_group()`  
`(nipyapi.nifi.apis.remote_process_groups_api.RemoteProcessGro`  
`method), 155`  
`get_remote_process_group_status()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 84`  
`get_remote_process_group_status_history()`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method), 85`  
`get_remote_process_group_status_history_with_http_`  
`(nipyapi.nifi.apis.flow_api.FlowApi`  
`method),`



```

85                                     get_runtime_manifest ()
get_remote_process_group_status_with_http_info ((nipyapi.nifi.apis.flow_api.FlowApi method),
(nipyapi.nifi.apis.flow_api.FlowApi method), 87
85                                     get_runtime_manifest_with_http_info ()
get_remote_process_group_with_http_info () (nipyapi.nifi.apis.flow_api.FlowApi method),
(nipyapi.nifi.apis.remote_process_groups_api.RemoteProcessGroupsApi
method), 156                                     get_search_options ()
get_remote_process_groups () (nipyapi.nifi.apis.provenance_api.ProvenanceApi
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi method), 151
method), 130                                     get_search_options_with_http_info ()
get_remote_process_groups_with_http_info () (nipyapi.nifi.apis.provenance_api.ProvenanceApi
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi method), 151
method), 130                                     get_service_access_status () (in module
get_replace_process_group_request () nipyapi.security), 22
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi method), 130                                     get_service_user () (in module nipyapi.security),
23
get_replace_process_group_request_with_http_info get_service_user_group () (in module
(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi nipyapi.security), 25
method), 131                                     get_site_to_site_details ()
get_reporting_task () (nipyapi.nifi.apis.site_to_site_api.SiteToSiteApi
(nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi method), 170
method), 164                                     get_site_to_site_details_with_http_info ()
get_reporting_task_types () (nipyapi.nifi.apis.site_to_site_api.SiteToSiteApi
(nipyapi.nifi.apis.flow_api.FlowApi method), method), 170
86                                     get_state () (nipyapi.nifi.apis.controller_services_api.ControllerService
method), 55
get_reporting_task_types_with_http_info () (nipyapi.nifi.apis.processors_api.ProcessorsApi
(nipyapi.nifi.apis.flow_api.FlowApi method), get_state () (nipyapi.nifi.apis.processors_api.ProcessorsApi
86 method), 145
get_reporting_task_with_http_info () get_state () (nipyapi.nifi.apis.remote_process_groups_api.RemoteProc
(nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi method), 156
method), 165                                     get_state () (nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi
method), 165
get_reporting_tasks () (nipyapi.nifi.apis.flow_api.FlowApi method), get_state_with_http_info ()
86 (nipyapi.nifi.apis.controller_services_api.ControllerServicesApi
method), 55
get_reporting_tasks_with_http_info () (nipyapi.nifi.apis.flow_api.FlowApi method), get_state_with_http_info ()
86 (nipyapi.nifi.apis.processors_api.ProcessorsApi
method), 145
get_resources () (nipyapi.nifi.apis.resources_api.ResourcesApi method), 145
method), 169                                     get_state_with_http_info ()
get_resources () (nipyapi.registry.apis.policies_api.PoliciesApi (nipyapi.nifi.apis.remote_process_groups_api.RemoteProcessGro
method), 561                                     method), 156
get_resources_with_http_info () get_state_with_http_info ()
(nipyapi.nifi.apis.resources_api.ResourcesApi (nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi
method), 169 method), 165
get_resources_with_http_info () get_system_diagnostics () (in module
(nipyapi.registry.apis.policies_api.PoliciesApi nipyapi.system), 25
method), 561                                     get_system_diagnostics ()
get_revert_request () (nipyapi.nifi.apis.system_diagnostics_api.SystemDiagnosticsApi
(nipyapi.nifi.apis.versions_api.VersionsApi method), 173
method), 184                                     get_system_diagnostics_with_http_info ()
get_revert_request_with_http_info () (nipyapi.nifi.apis.system_diagnostics_api.SystemDiagnosticsApi
(nipyapi.nifi.apis.versions_api.VersionsApi method), 173
method), 184                                     get_template () (in module nipyapi.templates), 27
get_root_pg_id () (in module nipyapi.canvas), 10 get_template_by_name () (in module

```

[nipyapi.templates](#)), 26  
[get\\_templates\(\)](#) ([nipyapi.nifi.apis.flow\\_api.FlowApi](#) method), 87  
[get\\_templates\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.flow\\_api.FlowApi](#) method), 87  
[get\\_test\\_url\\_status\(\)](#) ([nipyapi.utils.DockerContainer](#) method), 30  
[get\\_update\\_request\(\)](#) ([nipyapi.nifi.apis.versions\\_api.VersionsApi](#) method), 185  
[get\\_update\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.versions\\_api.VersionsApi](#) method), 185  
[get\\_user\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 176  
[get\\_user\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 564  
[get\\_user\\_group\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 176  
[get\\_user\\_group\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 564  
[get\\_user\\_group\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 176  
[get\\_user\\_group\\_with\\_http\\_info\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 564  
[get\\_user\\_groups\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 177  
[get\\_user\\_groups\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 564  
[get\\_user\\_groups\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 177  
[get\\_user\\_groups\\_with\\_http\\_info\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 565  
[get\\_user\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 177  
[get\\_user\\_with\\_http\\_info\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 565  
[get\\_users\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 177  
[get\\_users\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 565  
[get\\_users\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.tenants\\_api.TenantsApi](#) method), 177  
[get\\_users\\_with\\_http\\_info\(\)](#) ([nipyapi.registry.apis.tenants\\_api.TenantsApi](#) method), 565  
[get\\_validation\\_request\(\)](#) ([nipyapi.nifi.apis.parameter\\_contexts\\_api.ParameterContextsApi](#) method), 109  
[get\\_validation\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.parameter\\_contexts\\_api.ParameterContextsApi](#) method), 110  
[get\\_variable\\_registry\(\)](#) (in module [nipyapi.canvas](#)), 14  
[get\\_variable\\_registry\(\)](#) ([nipyapi.nifi.apis.process\\_groups\\_api.ProcessGroupsApi](#) method), 131  
[get\\_variable\\_registry\\_update\\_request\(\)](#) ([nipyapi.nifi.apis.process\\_groups\\_api.ProcessGroupsApi](#) method), 131  
[get\\_variable\\_registry\\_update\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.process\\_groups\\_api.ProcessGroupsApi](#) method), 131  
[get\\_variable\\_registry\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.process\\_groups\\_api.ProcessGroupsApi](#) method), 132  
[get\\_verification\\_request\(\)](#) ([nipyapi.nifi.apis.controller\\_services\\_api.ControllerServicesApi](#) method), 55  
[get\\_verification\\_request\(\)](#) ([nipyapi.nifi.apis.processors\\_api.ProcessorsApi](#) method), 145  
[get\\_verification\\_request\(\)](#) ([nipyapi.nifi.apis.reporting\\_tasks\\_api.ReportingTasksApi](#) method), 165  
[get\\_verification\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.controller\\_services\\_api.ControllerServicesApi](#) method), 56  
[get\\_verification\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.processors\\_api.ProcessorsApi](#) method), 146  
[get\\_verification\\_request\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.reporting\\_tasks\\_api.ReportingTasksApi](#) method), 166  
[get\\_version\\_info\(\)](#) (in module [nipyapi.versioning](#)), 33  
[get\\_version\\_information\(\)](#) ([nipyapi.nifi.apis.versions\\_api.VersionsApi](#) method), 185  
[get\\_version\\_information\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.versions\\_api.VersionsApi](#) method), 186  
[get\\_versions\(\)](#) ([nipyapi.nifi.apis.flow\\_api.FlowApi](#) method), 87  
[get\\_versions\\_with\\_http\\_info\(\)](#) ([nipyapi.nifi.apis.flow\\_api.FlowApi](#) method), 88

`getheader()` (*nipyapi.nifi.rest.RESTResponse* attribute), 536  
`getheader()` (*nipyapi.registry.rest.RESTResponse* attribute), 622  
`getheaders()` (*nipyapi.nifi.rest.RESTResponse* attribute), 536  
`getheaders()` (*nipyapi.registry.rest.RESTResponse* attribute), 622  
`getz_index(nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 240  
`getz_index(nipyapi.nifi.models.connection_entity.ConnectionEntity` attribute), 243  
`getz_index(nipyapi.nifi.models.label_dto.LabelDTO` attribute), 313  
`getz_index(nipyapi.nifi.models.label_entity.LabelEntity` attribute), 315  
`global_get_flow()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 554  
`global_get_flow_version()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 555  
`global_get_flow_version_with_http_info()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 555  
`global_get_flow_versions()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 555  
`global_get_flow_versions_with_http_info()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 556  
`global_get_flow_with_http_info()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 556  
`global_get_latest_flow_version()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 556  
`global_get_latest_flow_version_metadata()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 556  
`global_get_latest_flow_version_metadata_with_http_info()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 556  
`global_get_latest_flow_version_with_http_info()` (*nipyapi.registry.apis.flows\_api.FlowsApi* method), 557  
`group(nipyapi.nifi.models.bundle.Bundle` attribute), 222  
`group(nipyapi.nifi.models.bundle_dto.BundleDTO` attribute), 223  
`group(nipyapi.registry.models.bundle.Bundle` attribute), 575  
`group_access_control` (*nipyapi.nifi.models.port\_dto.PortDTO* attribute), 338  
`group_id(nipyapi.nifi.models.bulletin_dto.BulletinDTO` attribute), 219  
`group_id(nipyapi.nifi.models.bulletin_entity.BulletinEntity` attribute), 221  
`group_id(nipyapi.nifi.models.component_search_result_dto.ComponentSearchResultDTO` attribute), 233  
`group_id(nipyapi.nifi.models.connectable_component.ConnectableComponent` attribute), 236  
`group_id(nipyapi.nifi.models.connectable_dto.ConnectableDTO` attribute), 237  
`group_id(nipyapi.nifi.models.connection_status_dto.ConnectionStatusDTO` attribute), 245  
`group_id(nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO` attribute), 249  
`group_id(nipyapi.nifi.models.controller_service_referencing_component.ControllerServiceReferencingComponent` attribute), 268  
`group_id(nipyapi.nifi.models.port_status_dto.PortStatusDTO` attribute), 342  
`group_id(nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO` attribute), 345  
`group_id(nipyapi.nifi.models.processor_status_dto.ProcessorStatusDTO` attribute), 381  
`group_id(nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO` attribute), 385  
`group_id(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO` attribute), 398  
`group_id(nipyapi.nifi.models.remote_process_group_port_dto.RemoteProcessGroupPortDTO` attribute), 425  
`group_id(nipyapi.nifi.models.remote_process_group_status_dto.RemoteProcessGroupStatusDTO` attribute), 428  
`group_id(nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO` attribute), 432  
`group_id(nipyapi.nifi.models.template_dto.TemplateDTO` attribute), 465  
`group_id(nipyapi.nifi.models.version_control_information_dto.VersionControlInformationDTO` attribute), 490  
`group_id(nipyapi.registry.models.connectable_component.ConnectableComponent` attribute), 576  
`group_identifier(nipyapi.nifi.models.versioned_connection.VersionedConnection` attribute), 495  
`group_identifier(nipyapi.nifi.models.versioned_controller_service_referencing_component.VersionedControllerServiceReferencingComponent` attribute), 498  
`group_identifier(nipyapi.nifi.models.versioned_funnel.VersionedFunnel` attribute), 512  
`group_identifier(nipyapi.nifi.models.versioned_label.VersionedLabel` attribute), 513  
`group_identifier(nipyapi.nifi.models.versioned_port.VersionedPort` attribute), 516  
`group_identifier(nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup` attribute), 520  
`group_identifier(nipyapi.nifi.models.versioned_processor.VersionedProcessor` attribute), 523  
`group_identifier(nipyapi.nifi.models.versioned_remote_group_port.VersionedRemoteGroupPort` attribute), 523



[attribute](#)), 528  
[group\\_identifier \(nipyapi.nifi.models.versioned\\_remote\\_process\\_group.attribute\)](#), 531  
[group\\_identifier \(nipyapi.registry.models.versioned\\_connection.attribute\)](#), 587  
[group\\_identifier \(nipyapi.registry.models.versioned\\_controller.attribute\)](#), 591  
[group\\_identifier \(nipyapi.registry.models.versioned\\_funnel.attribute\)](#), 599  
[group\\_identifier \(nipyapi.registry.models.versioned\\_label.attribute\)](#), 600  
[group\\_identifier \(nipyapi.registry.models.versioned\\_port.attribute\)](#), 602  
[group\\_identifier \(nipyapi.registry.models.versioned\\_process\\_group.attribute\)](#), 606  
[group\\_identifier \(nipyapi.registry.models.versioned\\_processor.attribute\)](#), 609  
[group\\_identifier \(nipyapi.registry.models.versioned\\_remote\\_group.attribute\)](#), 614  
[group\\_identifier \(nipyapi.registry.models.versioned\\_remote\\_process\\_group.attribute\)](#), 617

## H

[HEAD \(\) \(nipyapi.nifi.rest.RESTClientObject method\)](#), 536  
[HEAD \(\) \(nipyapi.registry.rest.RESTClientObject method\)](#), 622  
[header\\_text \(nipyapi.nifi.models.banner\\_dto.BannerDTO attribute\)](#), 212  
[heap\\_utilization \(nipyapi.nifi.models.system\\_diagnostics\\_snapshot.attribute\)](#), 462  
[heartbeat \(nipyapi.nifi.models.node\\_dto.NodeDTO attribute\)](#), 325  
[height \(nipyapi.nifi.models.dimensions\\_dto.DimensionsDTO attribute\)](#), 286  
[height \(nipyapi.nifi.models.label\\_dto.LabelDTO attribute\)](#), 313  
[height \(nipyapi.nifi.models.versioned\\_label.VersionedLabel attribute\)](#), 514  
[height \(nipyapi.registry.models.versioned\\_label.VersionedLabel attribute\)](#), 600  
[history \(nipyapi.nifi.models.history\\_entity.HistoryEntity attribute\)](#), 310  
[HistoryDTO \(class in nipyapi.nifi.models.history\\_dto\)](#), 309  
[HistoryEntity \(class in nipyapi.nifi.models.history\\_entity\)](#), 310  
[hostname \(nipyapi.nifi.models.peer\\_dto.PeerDTO attribute\)](#), 335

[id \(nipyapi.nifi.models.access\\_policy\\_dto.AccessPolicyDTO attribute\)](#), 194  
[id \(nipyapi.nifi.models.access\\_policy\\_entity.AccessPolicyEntity attribute\)](#), 196  
[id \(nipyapi.nifi.models.access\\_policy\\_summary\\_dto.AccessPolicySummary attribute\)](#), 198  
[id \(nipyapi.nifi.models.access\\_policy\\_summary\\_entity.AccessPolicySummary attribute\)](#), 198  
[id \(nipyapi.nifi.models.action\\_dto.ActionDTO attribute\)](#), 200  
[id \(nipyapi.nifi.models.action\\_entity.ActionEntity attribute\)](#), 200  
[id \(nipyapi.nifi.models.activate\\_controller\\_services\\_entity.ActivateControllerServicesEntity attribute\)](#), 200  
[id \(nipyapi.nifi.models.affected\\_component\\_dto.AffectedComponentDTO attribute\)](#), 205  
[id \(nipyapi.nifi.models.affected\\_component\\_entity.AffectedComponentEntity attribute\)](#), 206  
[id \(nipyapi.nifi.models.bucket\\_dto.BucketDTO attribute\)](#), 208  
[id \(nipyapi.nifi.models.bucket\\_entity.BucketEntity attribute\)](#), 208  
[id \(nipyapi.nifi.models.bulletin\\_dto.BulletinDTO attribute\)](#), 220  
[id \(nipyapi.nifi.models.bulletin\\_entity.BulletinEntity attribute\)](#), 221  
[id \(nipyapi.nifi.models.component\\_reference\\_dto.ComponentReferenceDTO attribute\)](#), 230  
[id \(nipyapi.nifi.models.component\\_reference\\_entity.ComponentReferenceEntity attribute\)](#), 232  
[id \(nipyapi.nifi.models.component\\_search\\_result\\_dto.ComponentSearchResultDTO attribute\)](#), 233  
[id \(nipyapi.nifi.models.connectable\\_component.ConnectableComponent attribute\)](#), 236  
[id \(nipyapi.nifi.models.connectable\\_dto.ConnectableDTO attribute\)](#), 237  
[id \(nipyapi.nifi.models.connection\\_dto.ConnectionDTO attribute\)](#), 240  
[id \(nipyapi.nifi.models.connection\\_entity.ConnectionEntity attribute\)](#), 243  
[id \(nipyapi.nifi.models.connection\\_status\\_dto.ConnectionStatusDTO attribute\)](#), 245  
[id \(nipyapi.nifi.models.connection\\_status\\_snapshot\\_dto.ConnectionStatusSnapshotDTO attribute\)](#), 249  
[id \(nipyapi.nifi.models.connection\\_status\\_snapshot\\_entity.ConnectionStatusSnapshotEntity attribute\)](#), 252  
[id \(nipyapi.nifi.models.controller\\_dto.ControllerDTO attribute\)](#), 256  
[id \(nipyapi.nifi.models.controller\\_service\\_dto.ControllerServiceDTO attribute\)](#), 263  
[id \(nipyapi.nifi.models.controller\\_service\\_entity.ControllerServiceEntity attribute\)](#), 266  
[id \(nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_dto.ControllerServiceReferencingComponentDTO attribute\)](#), 268  
[id \(nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_entity.ControllerServiceReferencingComponentEntity attribute\)](#), 270

- `id (nipyapi.nifi.models.counter_dto.CounterDTO attribute), 278`
- `id (nipyapi.nifi.models.drop_request_dto.DropRequestDTO attribute), 290`
- `id (nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO attribute), 292`
- `id (nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity attribute), 293`
- `id (nipyapi.nifi.models.funnel_dto.FunnelDTO attribute), 305`
- `id (nipyapi.nifi.models.funnel_entity.FunnelEntity attribute), 307`
- `id (nipyapi.nifi.models.label_dto.LabelDTO attribute), 313`
- `id (nipyapi.nifi.models.label_entity.LabelEntity attribute), 315`
- `id (nipyapi.nifi.models.lineage_dto.LineageDTO attribute), 317`
- `id (nipyapi.nifi.models.listing_request_dto.ListingRequestDTO attribute), 321`
- `id (nipyapi.nifi.models.node_search_result_dto.NodeSearchResultDTO attribute), 332`
- `id (nipyapi.nifi.models.port_dto.PortDTO attribute), 338`
- `id (nipyapi.nifi.models.port_entity.PortEntity attribute), 341`
- `id (nipyapi.nifi.models.port_status_dto.PortStatusDTO attribute), 342`
- `id (nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO attribute), 345`
- `id (nipyapi.nifi.models.port_status_snapshot_entity.PortStatusSnapshotEntity attribute), 346`
- `id (nipyapi.nifi.models.process_group_dto.ProcessGroupDTO attribute), 352`
- `id (nipyapi.nifi.models.process_group_entity.ProcessGroupEntity attribute), 357`
- `id (nipyapi.nifi.models.process_group_flow_dto.ProcessGroupFlowDTO attribute), 361`
- `id (nipyapi.nifi.models.process_group_status_dto.ProcessGroupStatusDTO attribute), 363`
- `id (nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO attribute), 367`
- `id (nipyapi.nifi.models.process_group_status_snapshot_entity.ProcessGroupStatusSnapshotEntity attribute), 370`
- `id (nipyapi.nifi.models.processor_dto.ProcessorDTO attribute), 377`
- `id (nipyapi.nifi.models.processor_entity.ProcessorEntity attribute), 380`
- `id (nipyapi.nifi.models.processor_status_dto.ProcessorStatusDTO attribute), 382`
- `id (nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO attribute), 385`
- `id (nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity attribute), 387`
- `id (nipyapi.nifi.models.provenance_dto.ProvenanceDTO attribute), 394`
- `id (nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO attribute), 399`
- `id (nipyapi.nifi.models.provenance_node_dto.ProvenanceNodeDTO attribute), 404`
- `id (nipyapi.nifi.models.provenance_searchable_field_dto.ProvenanceSearchableFieldDTO attribute), 410`
- `id (nipyapi.nifi.models.registry_client_entity.RegistryClientEntity attribute), 412`
- `id (nipyapi.nifi.models.registry_dto.RegistryDTO attribute), 414`
- `id (nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO attribute), 418`
- `id (nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity attribute), 422`
- `id (nipyapi.nifi.models.remote_process_group_port_dto.RemoteProcessGroupPortDTO attribute), 425`
- `id (nipyapi.nifi.models.remote_process_group_port_entity.RemoteProcessGroupPortEntity attribute), 427`
- `id (nipyapi.nifi.models.remote_process_group_status_dto.RemoteProcessGroupStatusDTO attribute), 429`
- `id (nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO attribute), 432`
- `id (nipyapi.nifi.models.remote_process_group_status_snapshot_entity.RemoteProcessGroupStatusSnapshotEntity attribute), 433`
- `id (nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO attribute), 436`
- `id (nipyapi.nifi.models.reporting_task_entity.ReportingTaskEntity attribute), 439`
- `id (nipyapi.nifi.models.schedule_components_entity.ScheduleComponentsEntity attribute), 444`
- `id (nipyapi.nifi.models.snippet_dto.SnippetDTO attribute), 448`
- `id (nipyapi.nifi.models.template_dto.TemplateDTO attribute), 465`
- `id (nipyapi.nifi.models.template_entity.TemplateEntity attribute), 467`
- `id (nipyapi.nifi.models.tenant_dto.TenantDTO attribute), 469`
- `id (nipyapi.nifi.models.tenant_entity.TenantEntity attribute), 470`
- `id (nipyapi.nifi.models.tenant_status_dto.TenantStatusDTO attribute), 473`
- `id (nipyapi.nifi.models.tenant_status_snapshot_dto.TenantStatusSnapshotDTO attribute), 473`
- `id (nipyapi.nifi.models.tenant_status_snapshot_entity.TenantStatusSnapshotEntity attribute), 473`
- `id (nipyapi.nifi.models.user_dto.UserDTO attribute), 474`
- `id (nipyapi.nifi.models.user_entity.UserEntity attribute), 476`
- `id (nipyapi.nifi.models.user_group_dto.UserGroupDTO attribute), 477`
- `id (nipyapi.nifi.models.user_group_entity.UserGroupEntity attribute), 479`
- `id (nipyapi.nifi.models.user_group_status_dto.UserGroupStatusDTO attribute), 479`
- `id (nipyapi.nifi.models.user_group_status_snapshot_dto.UserGroupStatusSnapshotDTO attribute), 479`
- `id (nipyapi.nifi.models.user_group_status_snapshot_entity.UserGroupStatusSnapshotEntity attribute), 479`
- `id (nipyapi.nifi.models.connectable_component.ConnectableComponent attribute), 576`
- `identifier (nipyapi.nifi.models.bucket.Bucket attribute), 576`

tribute), 215

identifier (nipyapi.nifi.models.resource\_dto.ResourceDTO attribute), 442

identifier (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO attribute), 456

identifier (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 495

identifier (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService attribute), 498

identifier (nipyapi.nifi.models.versioned\_flow.VersionedFlow attribute), 500

identifier (nipyapi.nifi.models.versioned\_funnel.VersionedFunnel attribute), 512

identifier (nipyapi.nifi.models.versioned\_label.VersionedLabel attribute), 514

identifier (nipyapi.nifi.models.versioned\_port.VersionedPort attribute), 516

identifier (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520

identifier (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 524

identifier (nipyapi.nifi.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort attribute), 528

identifier (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 531

identifier (nipyapi.registry.models.access\_policy.AccessPolicy attribute), 569

identifier (nipyapi.registry.models.access\_policy\_summary.AccessPolicySummary attribute), 570

identifier (nipyapi.registry.models.bucket.Bucket attribute), 572

identifier (nipyapi.registry.models.bucket\_item.BucketItem attribute), 574

identifier (nipyapi.registry.models.resource.Resource attribute), 580

identifier (nipyapi.registry.models.tenant.Tenant attribute), 582

identifier (nipyapi.registry.models.user.User attribute), 583

identifier (nipyapi.registry.models.user\_group.UserGroup attribute), 584

identifier (nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 587

identifier (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService attribute), 591

identifier (nipyapi.registry.models.versioned\_flow.VersionedFlow attribute), 593

identifier (nipyapi.registry.models.versioned\_funnel.VersionedFunnel attribute), 599

identifier (nipyapi.registry.models.versioned\_label.VersionedLabel attribute), 600

identifier (nipyapi.registry.models.versioned\_port.VersionedPort attribute), 602

identifier (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606

identifier (nipyapi.registry.models.versioned\_processor.VersionedProcessor attribute), 610

identifier (nipyapi.registry.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort attribute), 614

identifier (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 617

identifier (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService attribute), 391

identifies\_controller\_service (nipyapi.nifi.models.versioned\_property\_descriptor.VersionedPropertyDescriptor attribute), 526

identifies\_controller\_service (nipyapi.registry.models.versioned\_property\_descriptor.VersionedPropertyDescriptor attribute), 612

identifies\_controller\_service\_bundle (nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDto attribute), 391

identifies\_controller\_service\_bundle (nipyapi.nifi.models.access\_status\_dto.AccessStatusDTO attribute), 201

identifies\_controller\_service\_bundle (nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity attribute), 285

identifies\_controller\_service\_bundle (nipyapi.nifi.models.tenant\_dto.TenantDTO attribute), 469

identifies\_controller\_service\_bundle (nipyapi.nifi.models.user\_dto.UserDTO attribute), 474

identifies\_controller\_service\_bundle (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO attribute), 477

identity (nipyapi.registry.models.current\_user.CurrentUser attribute), 578

identity (nipyapi.registry.models.tenant.Tenant attribute), 582

identity (nipyapi.registry.models.user.User attribute), 583

identity (nipyapi.registry.models.user\_group.UserGroup attribute), 584

import\_flow\_version() (in module nipyapi.versioning), 34

import\_process\_group() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 132

import\_process\_group\_with\_http\_info() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 133

import\_template() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 133

import\_template\_with\_http\_info() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 133

import\_versioned\_flow() (nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi method), 49

`import_versioned_flow_with_http_info()` *attribute*), 399  
`(nipyapi.registry.apis.bucket_flows_api.BucketFlowsApi`  
`method)`, 550 *input\_content\_claim\_container*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_input_port_count` *attribute*), 399  
`(nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO`  
`attribute)`, 419 *input\_content\_claim\_file\_size*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_output_port_count` *attribute*), 399  
`(nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO`  
`attribute)`, 419 *input\_content\_claim\_file\_size\_bytes*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_port_count` *attribute*), 399  
`(nipyapi.nifi.models.controller_dto.ControllerDTO`  
`attribute)`, 256 *input\_content\_claim\_identifier*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_port_count` *attribute*), 399  
`(nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO`  
`attribute)`, 275 *input\_content\_claim\_offset*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_port_count` *attribute*), 399  
`(nipyapi.nifi.models.process_group_dto.ProcessGroupDTO`  
`attribute)`, 352 *input\_content\_claim\_section*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 399  
`inactive_remote_port_count` *attribute*), 399  
`(nipyapi.nifi.models.process_group_entity.ProcessGroupEntity`  
`attribute)`, 357 *input\_port\_count* (*nipyapi.nifi.models.controller\_dto.ControllerDTO*  
`attribute)`, 256  
`incremental_results` *input\_port\_count* (*nipyapi.nifi.models.process\_group\_dto.ProcessGr*  
`(nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDTO`  
`attribute)`, 407 *input\_port\_count* (*nipyapi.nifi.models.process\_group\_entity.ProcessG*  
`attribute)`, 357  
`infer_object_label_from_class()` (*in mod-*  
`ule nipyapi.utils`), 30 *input\_port\_count* (*nipyapi.nifi.models.remote\_process\_group\_dto.Re*  
`attribute)`, 419  
`initiate_replace_process_group()` *input\_port\_count* (*nipyapi.nifi.models.remote\_process\_group\_entity.I*  
`(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
`method)`, 133 *attribute), 423  
`initiate_replace_process_group_with_http_info()` *input\_port\_results*  
`(nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
`method)`, 133 *(nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO*  
`attribute)`, 445  
`initiate_revert_flow_version()` *input\_port\_status\_snapshots*  
`(nipyapi.nifi.apis.versions_api.VersionsApi`  
`method)`, 186 *(nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessG*  
`attribute)`, 368  
`initiate_revert_flow_version_with_http_info()` *input\_ports* (*nipyapi.nifi.models.controller\_dto.ControllerDTO*  
`(nipyapi.nifi.apis.versions_api.VersionsApi`  
`method)`, 186 *attribute), 257  
`initiate_version_control_update()` *input\_ports* (*nipyapi.nifi.models.flow\_dto.FlowDTO*  
`(nipyapi.nifi.apis.versions_api.VersionsApi`  
`method)`, 187 *attribute), 297  
`initiate_version_control_update_with_http_info()` *input\_ports* (*nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO*  
`(nipyapi.nifi.apis.versions_api.VersionsApi`  
`method)`, 187 *attribute), 304  
`input` (*nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO*  
`attribute)`, 250 *input\_ports* (*nipyapi.nifi.models.input\_ports\_entity.InputPortsEntity*  
`attribute)`, 311  
`input` (*nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO*  
`attribute)`, 345 *input\_ports* (*nipyapi.nifi.models.remote\_process\_group\_contents\_dto.R*  
`attribute)`, 448 *input\_ports* (*nipyapi.nifi.models.snippet\_dto.SnippetDTO*  
`attribute)`, 520  
`input` (*nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO*  
`attribute)`, 368 *input\_ports* (*nipyapi.nifi.models.versioned\_process\_group.VersionedPr*  
`attribute)`, 531  
`input` (*nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO*  
`attribute)`, 385 *input\_ports* (*nipyapi.nifi.models.versioned\_remote\_process\_group.Vers*  
`attribute)`, 531  
`input_content_available` *input\_ports* (*nipyapi.registry.models.versioned\_process\_group.Version*  
`(nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO`  
`attribute)`, 606 *(nipyapi.registry.models.versioned\_remote\_process\_group*****



attribute), 617  
 input\_requirement (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 377  
 input\_requirement (nipyapi.nifi.models.processor\_entity.ProcessorEntity attribute), 380  
 InputPortsApi (class in nipyapi.nifi.apis.input\_ports\_api), 98  
 InputPortsEntity (class in nipyapi.nifi.models.input\_ports\_entity), 311  
 instance\_id (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 257  
 instance\_identifier (nipyapi.nifi.models.connectable\_component.ConnectableComponent attribute), 236  
 instance\_identifier (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 495  
 instance\_identifier (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService attribute), 498  
 instance\_identifier (nipyapi.nifi.models.versioned\_funnel.VersionedFunnel attribute), 512  
 instance\_identifier (nipyapi.nifi.models.versioned\_label.VersionedLabel attribute), 514  
 instance\_identifier (nipyapi.nifi.models.versioned\_port.VersionedPort attribute), 516  
 instance\_identifier (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520  
 instance\_identifier (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 524  
 instance\_identifier (nipyapi.nifi.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort attribute), 528  
 instance\_identifier (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 531  
 instance\_identifier (nipyapi.registry.models.connectable\_component.ConnectableComponent attribute), 576  
 instance\_identifier (nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 587  
 instance\_identifier (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService attribute), 591  
 instance\_identifier (nipyapi.registry.models.versioned\_funnel.VersionedFunnel attribute), 599  
 instance\_identifier (nipyapi.registry.models.versioned\_label.VersionedLabel attribute), 600  
 instance\_identifier (nipyapi.registry.models.versioned\_port.VersionedPort attribute), 603  
 instance\_identifier (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606  
 instance\_identifier (nipyapi.registry.models.versioned\_processor.VersionedProcessor attribute), 610  
 instance\_identifier (nipyapi.registry.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort attribute), 614  
 instance\_identifier (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 617  
 instantiate\_template() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 134  
 instantiate\_template\_with\_http\_info() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 134  
 InstantiateTemplateRequestEntity (class in nipyapi.nifi.models.instantiate\_template\_request\_entity), 311  
 invalid\_count (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 257  
 invalid\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO attribute), 275  
 invalid\_count (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 352  
 invalid\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 357  
 is\_endpoint\_up() (in module nipyapi.utils), 29  
 ItemsApi (class in nipyapi.registry.apis.items\_api), 477  
 J  
 java\_version (nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO attribute), 492  
 K  
 key (nipyapi.nifi.models.state\_entry\_dto.StateEntryDTO attribute), 451  
 knox\_callback() (nipyapi.nifi.apis.access\_api.AccessApi method), 38  
 knox\_callback\_with\_http\_info() (nipyapi.nifi.apis.access\_api.AccessApi method), 39

knox\_logout() (nipyapi.nifi.apis.access\_api.AccessApi last\_refreshed(nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO attribute), 39  
 knox\_logout\_with\_http\_info() last\_updated(nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO attribute), 290  
 (nipyapi.nifi.apis.access\_api.AccessApi attribute), 290  
 knox\_request() (nipyapi.nifi.apis.access\_api.AccessApi last\_updated(nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO attribute), 321  
 method), 39 last\_updated(nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO attribute), 485  
 knox\_request\_with\_http\_info() last\_updated(nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO attribute), 509  
 (nipyapi.nifi.apis.access\_api.AccessApi attribute), 39 latest(nipyapi.nifi.models.versioned\_flow\_coordinates.VersionedFlowCoordinates attribute), 502  
**L** latest(nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot attribute), 505  
 label (nipyapi.nifi.models.label\_dto.LabelDTO attribute), 313 latest(nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot attribute), 596  
 label (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO attribute), 410 latest(nipyapi.registry.models.versioned\_flow\_snapshot.VersionedFlowSnapshot attribute), 596  
 label (nipyapi.nifi.models.status\_descriptor\_dto.StatusDescriptorDTO attribute), 453 level (nipyapi.nifi.models.bulletin\_dto.BulletinDTO attribute), 220  
 label (nipyapi.nifi.models.versioned\_label.VersionedLabel attribute), 514 lineage (nipyapi.nifi.models.lineage\_entity.LineageEntity attribute), 318  
 label (nipyapi.registry.models.versioned\_label.VersionedLabel attribute), 601 lineage\_duration(nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 300  
 label\_index(nipyapi.nifi.models.connection\_dto.ConnectionDTO attribute), 240 lineage\_duration(nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 303  
 label\_index(nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 243 lineage\_entity(nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 400  
 label\_index(nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 495 lineage\_request(nipyapi.nifi.models.lineage\_request\_dto.LineageRequestDTO attribute), 319  
 label\_index(nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 587 lineage\_results(nipyapi.nifi.models.lineage\_results\_dto.LineageResultsDTO attribute), 319  
 label\_results(nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO attribute), 446 LineageDTO (class in nipyapi.nifi.models.lineage\_dto), 316  
 LabelDTO (class in nipyapi.nifi.models.label\_dto), 313 LineageEntity (class in nipyapi.nifi.models.lineage\_entity), 317  
 LabelEntity (class in nipyapi.nifi.models.label\_entity), 314 LineageRequestDTO (class in nipyapi.nifi.models.lineage\_request\_dto), 318  
 labels (nipyapi.nifi.models.flow\_dto.FlowDTO attribute), 297 LineageResultsDTO (class in nipyapi.nifi.models.lineage\_results\_dto), 319  
 labels (nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO attribute), 304 link (nipyapi.nifi.models.bucket.Bucket attribute), 215  
 labels (nipyapi.nifi.models.labels\_entity.LabelsEntity attribute), 316 link (nipyapi.nifi.models.versioned\_flow.VersionedFlow attribute), 500  
 labels (nipyapi.nifi.models.snippet\_dto.SnipppetDTO attribute), 448 link (nipyapi.registry.models.bucket.Bucket attribute), 572  
 labels (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520 link (nipyapi.registry.models.bucket\_item.BucketItem attribute), 574  
 labels (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606 link (nipyapi.registry.models.versioned\_flow.VersionedFlow attribute), 593  
 LabelsApi (class in nipyapi.nifi.apis.labels\_api), 100 link (nipyapi.registry.models.versioned\_flow\_snapshot\_metadata.VersionedFlowSnapshotMetadata attribute), 598  
 LabelsEntity (class in nipyapi.nifi.models.labels\_entity), 316 link\_dto(nipyapi.nifi.models.lineage\_results\_dto.LineageResultsDTO attribute), 319  
 last\_modifier(nipyapi.nifi.models.revision\_dto.RevisionDTO attribute), 443  
 last\_refreshed(nipyapi.nifi.models.history\_dto.HistoryDTO attribute), 309

---

`list_all_by_kind()` (in module `nipyapi.canvas`), 17  
`list_all_connections()` (in module `nipyapi.canvas`), 15  
`list_all_controller_types()` (in module `nipyapi.canvas`), 17  
`list_all_controllers()` (in module `nipyapi.canvas`), 16  
`list_all_funnels()` (in module `nipyapi.canvas`), 18  
`list_all_input_ports()` (in module `nipyapi.canvas`), 17  
`list_all_output_ports()` (in module `nipyapi.canvas`), 18  
`list_all_parameter_contexts()` (in module `nipyapi.parameters`), 19  
`list_all_process_groups()` (in module `nipyapi.canvas`), 11  
`list_all_processor_types()` (in module `nipyapi.canvas`), 12  
`list_all_processors()` (in module `nipyapi.canvas`), 12  
`list_all_remote_process_groups()` (in module `nipyapi.canvas`), 18  
`list_all_templates()` (in module `nipyapi.templates`), 26  
`list_flow_versions()` (in module `nipyapi.versioning`), 35  
`list_flows_in_bucket()` (in module `nipyapi.versioning`), 33  
`list_invalid_processors()` (in module `nipyapi.canvas`), 15  
`list_registry_buckets()` (in module `nipyapi.versioning`), 32  
`list_registry_clients()` (in module `nipyapi.versioning`), 31  
`list_sensitive_processors()` (in module `nipyapi.canvas`), 15  
`list_service_user_groups()` (in module `nipyapi.security`), 25  
`list_service_users()` (in module `nipyapi.security`), 23  
`listing_request` (`nipyapi.nifi.models.listing_request_entity.ListingRequestEntity` attribute), 322  
`ListingRequestDTO` (class in `nipyapi.nifi.models.listing_request_dto`), 320  
`ListingRequestEntity` (class in `nipyapi.nifi.models.listing_request_entity`), 322  
`load()` (in module `nipyapi.utils`), 28  
`load_balance_compression` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 240  
`load_balance_compression` (`nipyapi.nifi.models.versioned_connection.VersionedConnection` attribute), 495  
`load_balance_compression` (`nipyapi.registry.models.versioned_connection.VersionedConnection` attribute), 588  
`load_balance_partition_attribute` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 240  
`load_balance_status` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 240  
`load_balance_strategy` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 240  
`load_balance_strategy` (`nipyapi.nifi.models.versioned_connection.VersionedConnection` attribute), 496  
`load_balance_strategy` (`nipyapi.registry.models.versioned_connection.VersionedConnection` attribute), 588  
`load_template_from_xml_file_path()` (in module `nipyapi.templates`), 27  
`load_template_from_xml_file_stream()` (in module `nipyapi.templates`), 28  
`load_template_from_xml_string()` (in module `nipyapi.templates`), 28  
`local_input_port_count` (`nipyapi.nifi.models.process_group_dto.ProcessGroupDTO` attribute), 352  
`local_input_port_count` (`nipyapi.nifi.models.process_group_entity.ProcessGroupEntity` attribute), 358  
`local_network_interface` (`nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO` attribute), 419  
`local_network_interface` (`nipyapi.nifi.models.versioned_remote_process_group.VersionedRemoteProcessGroup` attribute), 531  
`local_network_interface` (`nipyapi.registry.models.versioned_remote_process_group.VersionedRemoteProcessGroup` attribute), 617  
`local_output_port_count` (`nipyapi.nifi.models.process_group_dto.ProcessGroupDTO` attribute), 352  
`local_output_port_count` (`nipyapi.nifi.models.process_group_entity.ProcessGroupEntity` attribute), 358  
`local_state` (`nipyapi.nifi.models.component_state_dto.ComponentStateDTO` attribute), 235  
`locally_modified_and_stale_count` (`nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO` attribute), 275  
`locally_modified_and_stale_count`

(nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO, heap (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 352  
 attribute), 352  
 locally\_modified\_and\_stale\_count max\_non\_heap\_bytes (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 463  
 locally\_modified\_count max\_results (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO, attribute), 321  
 (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO, attribute), 275  
 max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 locally\_modified\_count max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO, attribute), 352  
 max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 locally\_modified\_count max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 (nipyapi.nifi.models.controller\_configuration\_dto.ControllerConfigurationDTO, attribute), 254  
 locally\_modified\_count max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity, attribute), 358  
 max\_results (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 log\_out () (nipyapi.nifi.apis.access\_api.AccessApi, attribute), 40  
 message (nipyapi.nifi.models.access\_status\_dto.AccessStatusDTO, attribute), 201  
 log\_out () (nipyapi.registry.apis.access\_api.AccessApi, attribute), 540  
 message (nipyapi.nifi.models.bulletin\_dto.BulletinDTO, attribute), 220  
 log\_out\_complete () (nipyapi.nifi.apis.access\_api.AccessApi, attribute), 40  
 message (nipyapi.nifi.models.node\_event\_dto.NodeEventDTO, attribute), 327  
 log\_out\_complete\_with\_http\_info () (nipyapi.nifi.apis.access\_api.AccessApi, attribute), 40  
 message (nipyapi.nifi.models.transaction\_result\_entity.TransactionResultEntity, attribute), 472  
 log\_out\_with\_http\_info () (nipyapi.nifi.apis.access\_api.AccessApi, attribute), 40  
 millis (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO, attribute), 403  
 millis (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO, attribute), 404  
 log\_out\_with\_http\_info () (nipyapi.registry.apis.access\_api.AccessApi, attribute), 540  
 minimum\_file\_size (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO, attribute), 407  
 login\_supported (nipyapi.registry.models.current\_user.CurrentUser, attribute), 578  
 modified\_timestamp (nipyapi.nifi.models.versioned\_flow.VersionedFlow, attribute), 593  
 loss\_tolerant (nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO, attribute), 374  
 modified\_timestamp (nipyapi.registry.models.bucket\_item.BucketItem, attribute), 574

## M

matches (nipyapi.nifi.models.component\_search\_result\_dto.ComponentSearchResultDTO, attribute), 233  
 matches (nipyapi.registry.models.versioned\_flow.VersionedFlow, attribute), 593  
 max\_backoff\_period (nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO, attribute), 374  
 multiple\_versions\_available (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO, attribute), 263  
 max\_backoff\_period (nipyapi.nifi.models.versioned\_processor.VersionedProcessor, attribute), 524  
 multiple\_versions\_available (nipyapi.nifi.models.processor\_dto.ProcessorDTO, attribute), 377  
 max\_backoff\_period (nipyapi.registry.models.versioned\_processor.VersionedProcessor, attribute), 610  
 multiple\_versions\_available (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO, attribute), 436  
 max\_event\_driven\_thread\_count (nipyapi.nifi.models.controller\_configuration\_dto.ControllerConfigurationDTO, attribute), 254

## N

max\_heap (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 462  
 max\_heap\_bytes (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 463  
 max\_heap (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 206  
 max\_heap\_bytes (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO, attribute), 211



name (nipyapi.nifi.models.bucket.Bucket attribute), 215  
 name (nipyapi.nifi.models.bucket\_dto.BucketDTO attribute), 216  
 name (nipyapi.nifi.models.component\_reference\_dto.ComponentReferenceDTO attribute), 230  
 name (nipyapi.nifi.models.component\_search\_result\_dto.ComponentSearchResultDTO attribute), 233  
 name (nipyapi.nifi.models.connectable\_component.ConnectableComponent attribute), 236  
 name (nipyapi.nifi.models.connectable\_dto.ConnectableDTO attribute), 238  
 name (nipyapi.nifi.models.connection\_dto.ConnectionDTO attribute), 241  
 name (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO attribute), 245  
 name (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO attribute), 250  
 name (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 257  
 name (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO attribute), 263  
 name (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO attribute), 268  
 name (nipyapi.nifi.models.counter\_dto.CounterDTO attribute), 278  
 name (nipyapi.nifi.models.create\_template\_request\_entity.CreateTemplateRequestEntity attribute), 283  
 name (nipyapi.nifi.models.flow\_breadcrumb\_dto.FlowBreadcrumbDTO attribute), 292  
 name (nipyapi.nifi.models.garbage\_collection\_dto.GarbageCollectionDTO attribute), 309  
 name (nipyapi.nifi.models.port\_dto.PortDTO attribute), 339  
 name (nipyapi.nifi.models.port\_status\_dto.PortStatusDTO attribute), 342  
 name (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO attribute), 345  
 name (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 353  
 name (nipyapi.nifi.models.process\_group\_status\_dto.ProcessGroupStatusDTO attribute), 363  
 name (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 368  
 name (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 377  
 name (nipyapi.nifi.models.processor\_status\_dto.ProcessorStatusDTO attribute), 382  
 name (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 385  
 name (nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDTO attribute), 391  
 name (nipyapi.nifi.models.registry\_dto.RegistryDTO attribute), 414  
 name (nipyapi.nifi.models.relationship\_dto.RelationshipDTO attribute), 415  
 name (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 419  
 name (nipyapi.nifi.models.remote\_process\_group\_port\_dto.RemoteProcessGroupPortDTO attribute), 425  
 name (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO attribute), 429  
 name (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshotDTO attribute), 432  
 name (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 437  
 name (nipyapi.nifi.models.resource\_dto.ResourceDTO attribute), 442  
 name (nipyapi.nifi.models.template\_dto.TemplateDTO attribute), 466  
 name (nipyapi.nifi.models.variable\_dto.VariableDTO attribute), 481  
 name (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 496  
 name (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService attribute), 499  
 name (nipyapi.nifi.models.versioned\_controller\_service\_referencing\_component\_dto.VersionedControllerServiceReferencingComponentDTO attribute), 501  
 name (nipyapi.nifi.models.versioned\_funnel.VersionedFunnel attribute), 512  
 name (nipyapi.nifi.models.versioned\_label.VersionedLabel attribute), 514  
 name (nipyapi.nifi.models.versioned\_port.VersionedPort attribute), 516  
 name (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520  
 name (nipyapi.nifi.models.versioned\_processor.VersionedProcessor attribute), 524  
 name (nipyapi.nifi.models.versioned\_property\_descriptor.VersionedPropertyDescriptor attribute), 526  
 name (nipyapi.nifi.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort attribute), 528  
 name (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup attribute), 531  
 name (nipyapi.registry.models.bucket.Bucket attribute), 572  
 name (nipyapi.registry.models.bucket\_item.BucketItem attribute), 574  
 name (nipyapi.registry.models.connectable\_component.ConnectableComponent attribute), 576  
 name (nipyapi.registry.models.resource.Resource attribute), 580  
 name (nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 588  
 name (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService attribute), 591  
 name (nipyapi.registry.models.versioned\_flow.VersionedFlow attribute), 593  
 name (nipyapi.registry.models.versioned\_funnel.VersionedFunnel attribute), 593

*attribute*), 599  
 name (*nipyapi.registry.models.versioned\_label.VersionedLabel* *attribute*), 148  
*attribute*), 601  
 name (*nipyapi.registry.models.versioned\_port.VersionedPort* *attribute*), 603  
 name (*nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup* *attribute*), 606  
 name (*nipyapi.registry.models.versioned\_processor.VersionedProcessor* *attribute*), 610  
 name (*nipyapi.registry.models.versioned\_property\_descriptor.VersionedPropertyDescriptor* *attribute*), 612  
 name (*nipyapi.registry.models.versioned\_remote\_group\_port.VersionedRemoteGroupPort* *attribute*), 614  
 name (*nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup* *attribute*), 617  
 NATIVE\_TYPES\_MAPPING  
     (*nipyapi.nifi.api\_client.ApiClient* *attribute*), 533  
 NATIVE\_TYPES\_MAPPING  
     (*nipyapi.registry.api\_client.ApiClient* *attribute*), 619  
 ni\_fi\_version (*nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO* *attribute*), 493  
 nipyapi.canvas (*module*), 10  
 nipyapi.config (*module*), 19  
 nipyapi.nifi.api\_client (*module*), 533  
 nipyapi.nifi.apis.access\_api (*module*), 36  
 nipyapi.nifi.apis.connections\_api (*module*), 41  
 nipyapi.nifi.apis.controller\_api (*module*), 43  
 nipyapi.nifi.apis.controller\_services\_api (*module*), 51  
 nipyapi.nifi.apis.counters\_api (*module*), 59  
 nipyapi.nifi.apis.data\_transfer\_api (*module*), 61  
 nipyapi.nifi.apis.flow\_api (*module*), 66  
 nipyapi.nifi.apis.flowfile\_queues\_api (*module*), 91  
 nipyapi.nifi.apis.funnel\_api (*module*), 96  
 nipyapi.nifi.apis.input\_ports\_api (*module*), 98  
 nipyapi.nifi.apis.labels\_api (*module*), 100  
 nipyapi.nifi.apis.output\_ports\_api (*module*), 102  
 nipyapi.nifi.apis.parameter\_contexts\_api (*module*), 105  
 nipyapi.nifi.apis.policies\_api (*module*), 112  
 nipyapi.nifi.apis.process\_groups\_api (*module*), 116  
 nipyapi.nifi.apis.processors\_api (*module*), 140  
 nipyapi.nifi.apis.provenance\_api (*module*), 148  
 nipyapi.nifi.apis.provenance\_events\_api (*module*), 153  
 nipyapi.nifi.apis.remote\_process\_groups\_api (*module*), 155  
 nipyapi.nifi.apis.reporting\_tasks\_api (*module*), 162  
 nipyapi.nifi.apis.resources\_api (*module*), 169  
 nipyapi.nifi.apis.site\_to\_site\_api (*module*), 170  
 nipyapi.nifi.apis.snippets\_api (*module*), 171  
 nipyapi.nifi.apis.system\_diagnostics\_api (*module*), 173  
 nipyapi.nifi.apis.templates\_api (*module*), 174  
 nipyapi.nifi.apis.tenants\_api (*module*), 175  
 nipyapi.nifi.apis.versions\_api (*module*), 176  
 nipyapi.nifi.configuration (*module*), 535  
 nipyapi.nifi.models.about\_dto (*module*), 190  
 nipyapi.nifi.models.about\_entity (*module*), 192  
 nipyapi.nifi.models.access\_configuration\_dto (*module*), 192  
 nipyapi.nifi.models.access\_configuration\_entity (*module*), 193  
 nipyapi.nifi.models.access\_policy\_dto (*module*), 194  
 nipyapi.nifi.models.access\_policy\_entity (*module*), 195  
 nipyapi.nifi.models.access\_policy\_summary\_dto (*module*), 197  
 nipyapi.nifi.models.access\_policy\_summary\_entity (*module*), 199  
 nipyapi.nifi.models.access\_status\_dto (*module*), 200  
 nipyapi.nifi.models.access\_status\_entity (*module*), 201  
 nipyapi.nifi.models.action\_details\_dto (*module*), 202  
 nipyapi.nifi.models.action\_dto (*module*), 202  
 nipyapi.nifi.models.action\_entity (*module*), 204  
 nipyapi.nifi.models.activate\_controller\_services\_entity (*module*), 205  
 nipyapi.nifi.models.affected\_component\_dto (*module*), 206  
 nipyapi.nifi.models.affected\_component\_entity

(module), 207  
 nipyapi.nifi.models.allowable\_value\_dto (module), 209  
 nipyapi.nifi.models.allowable\_value\_entity (module), 210  
 nipyapi.nifi.models.attribute\_dto (module), 210  
 nipyapi.nifi.models.banner\_dto (module), 211  
 nipyapi.nifi.models.banner\_entity (module), 212  
 nipyapi.nifi.models.batch\_settings\_dto (module), 212  
 nipyapi.nifi.models.batch\_size (module), 213  
 nipyapi.nifi.models.bucket (module), 214  
 nipyapi.nifi.models.bucket\_dto (module), 216  
 nipyapi.nifi.models.bucket\_entity (module), 216  
 nipyapi.nifi.models.buckets\_entity (module), 217  
 nipyapi.nifi.models.bulletin\_board\_dto (module), 218  
 nipyapi.nifi.models.bulletin\_board\_entity (module), 218  
 nipyapi.nifi.models.bulletin\_dto (module), 219  
 nipyapi.nifi.models.bulletin\_entity (module), 221  
 nipyapi.nifi.models.bundle (module), 222  
 nipyapi.nifi.models.bundle\_dto (module), 223  
 nipyapi.nifi.models.cluste\_summary\_entity (module), 223  
 nipyapi.nifi.models.cluster\_dto (module), 224  
 nipyapi.nifi.models.cluster\_entity (module), 225  
 nipyapi.nifi.models.cluster\_search\_results\_entity (module), 225  
 nipyapi.nifi.models.cluster\_summary\_dto (module), 226  
 nipyapi.nifi.models.component\_details\_dto (module), 227  
 nipyapi.nifi.models.component\_difference\_dto (module), 227  
 nipyapi.nifi.models.component\_history\_dto (module), 229  
 nipyapi.nifi.models.component\_history\_entity (module), 229  
 nipyapi.nifi.models.component\_reference\_dto (module), 230  
 nipyapi.nifi.models.component\_reference\_entity (module), 231  
 nipyapi.nifi.models.component\_search\_result\_dto (module), 233  
 nipyapi.nifi.models.component\_state\_dto (module), 234  
 nipyapi.nifi.models.component\_state\_entity (module), 235  
 nipyapi.nifi.models.connectable\_component (module), 236  
 nipyapi.nifi.models.connectable\_dto (module), 237  
 nipyapi.nifi.models.connection\_dto (module), 238  
 nipyapi.nifi.models.connection\_entity (module), 242  
 nipyapi.nifi.models.connection\_status\_dto (module), 244  
 nipyapi.nifi.models.connection\_status\_entity (module), 246  
 nipyapi.nifi.models.connection\_status\_snapshot\_dto (module), 247  
 nipyapi.nifi.models.connection\_status\_snapshot\_entity (module), 251  
 nipyapi.nifi.models.connections\_entity (module), 252  
 nipyapi.nifi.models.controller\_bulletins\_entity (module), 252  
 nipyapi.nifi.models.controller\_configuration\_dto (module), 253  
 nipyapi.nifi.models.controller\_configuration\_entity (module), 254  
 nipyapi.nifi.models.controller\_dto (module), 255  
 nipyapi.nifi.models.controller\_entity (module), 258  
 nipyapi.nifi.models.controller\_service\_api (module), 259  
 nipyapi.nifi.models.controller\_service\_api\_dto (module), 259  
 nipyapi.nifi.models.controller\_service\_dto (module), 260  
 nipyapi.nifi.models.controller\_service\_entity (module), 265  
 nipyapi.nifi.models.controller\_service\_referencing\_dto (module), 267  
 nipyapi.nifi.models.controller\_service\_referencing\_entity (module), 269  
 nipyapi.nifi.models.controller\_service\_referencing\_dto (module), 271  
 nipyapi.nifi.models.controller\_service\_types\_entity (module), 272  
 nipyapi.nifi.models.controller\_services\_entity (module), 272  
 nipyapi.nifi.models.controller\_status\_dto

(*module*), 273

nipyapi.nifi.models.controller\_status\_entity (*module*), 276

nipyapi.nifi.models.copy\_snippet\_request\_entity (*module*), 277

nipyapi.nifi.models.counter\_dto (*module*), 278

nipyapi.nifi.models.counter\_entity (*module*), 279

nipyapi.nifi.models.counters\_dto (*module*), 279

nipyapi.nifi.models.counters\_entity (*module*), 280

nipyapi.nifi.models.counters\_snapshot\_dto (*module*), 281

nipyapi.nifi.models.create\_active\_request\_entity (*module*), 281

nipyapi.nifi.models.create\_template\_request\_entity (*module*), 282

nipyapi.nifi.models.current\_user\_entity (*module*), 283

nipyapi.nifi.models.difference\_dto (*module*), 285

nipyapi.nifi.models.dimensions\_dto (*module*), 286

nipyapi.nifi.models.documented\_type\_dto (*module*), 287

nipyapi.nifi.models.drop\_request\_dto (*module*), 288

nipyapi.nifi.models.drop\_request\_entity (*module*), 291

nipyapi.nifi.models.flow\_breadcrumb\_dto (*module*), 292

nipyapi.nifi.models.flow\_breadcrumb\_entity (*module*), 293

nipyapi.nifi.models.flow\_comparison\_entity (*module*), 294

nipyapi.nifi.models.flow\_configuration\_dto (*module*), 294

nipyapi.nifi.models.flow\_configuration\_entity (*module*), 296

nipyapi.nifi.models.flow\_dto (*module*), 297

nipyapi.nifi.models.flow\_entity (*module*), 298

nipyapi.nifi.models.flow\_file\_dto (*module*), 298

nipyapi.nifi.models.flow\_file\_entity (*module*), 301

nipyapi.nifi.models.flow\_file\_summary\_dto (*module*), 302

nipyapi.nifi.models.flow\_snippet\_dto (*module*), 304

nipyapi.nifi.models.funnel\_dto (*module*), 305

nipyapi.nifi.models.funnel\_entity (*module*), 306

nipyapi.nifi.models.funnels\_entity (*module*), 307

nipyapi.nifi.models.garbage\_collection\_dto (*module*), 308

nipyapi.nifi.models.history\_dto (*module*), 309

nipyapi.nifi.models.history\_entity (*module*), 310

nipyapi.nifi.models.input\_ports\_entity (*module*), 310

nipyapi.nifi.models.instantiate\_template\_request\_entity (*module*), 311

nipyapi.nifi.models.label\_dto (*module*), 312

nipyapi.nifi.models.label\_entity (*module*), 314

nipyapi.nifi.models.labels\_entity (*module*), 315

nipyapi.nifi.models.lineage\_dto (*module*), 316

nipyapi.nifi.models.lineage\_entity (*module*), 317

nipyapi.nifi.models.lineage\_request\_dto (*module*), 318

nipyapi.nifi.models.lineage\_results\_dto (*module*), 319

nipyapi.nifi.models.listing\_request\_dto (*module*), 320

nipyapi.nifi.models.listing\_request\_entity (*module*), 322

nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto (*module*), 323

nipyapi.nifi.models.node\_counters\_snapshot\_dto (*module*), 324

nipyapi.nifi.models.node\_dto (*module*), 325

nipyapi.nifi.models.node\_entity (*module*), 326

nipyapi.nifi.models.node\_event\_dto (*module*), 327

nipyapi.nifi.models.node\_port\_status\_snapshot\_dto (*module*), 328

nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto (*module*), 329

nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto (*module*), 330

nipyapi.nifi.models.node\_remote\_process\_group\_status\_dto (*module*), 331

nipyapi.nifi.models.node\_search\_result\_dto (*module*), 332

nipyapi.nifi.models.node\_status\_snapshots\_dto (*module*), 332

nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto (*module*), 333

(module), 333  
 nipyapi.nifi.models.output\_ports\_entity (module), 334  
 nipyapi.nifi.models.peer\_dto (module), 335  
 nipyapi.nifi.models.peers\_entity (module), 336  
 nipyapi.nifi.models.permissions (module), 336  
 nipyapi.nifi.models.permissions\_dto (module), 337  
 nipyapi.nifi.models.port\_dto (module), 338  
 nipyapi.nifi.models.port\_entity (module), 340  
 nipyapi.nifi.models.port\_status\_dto (module), 342  
 nipyapi.nifi.models.port\_status\_entity (module), 343  
 nipyapi.nifi.models.port\_status\_snapshot\_dto (module), 344  
 nipyapi.nifi.models.port\_status\_snapshot\_entity (module), 346  
 nipyapi.nifi.models.position\_dto (module), 347  
 nipyapi.nifi.models.previous\_value\_dto (module), 347  
 nipyapi.nifi.models.prioritizer\_types\_entity (module), 348  
 nipyapi.nifi.models.process\_group\_dto (module), 349  
 nipyapi.nifi.models.process\_group\_entity (module), 355  
 nipyapi.nifi.models.process\_group\_flow\_dto (module), 360  
 nipyapi.nifi.models.process\_group\_flow\_entity (module), 362  
 nipyapi.nifi.models.process\_group\_status\_dto (module), 362  
 nipyapi.nifi.models.process\_group\_status\_entity (module), 363  
 nipyapi.nifi.models.process\_group\_status\_snapshot\_dto (module), 364  
 nipyapi.nifi.models.process\_group\_status\_snapshot\_entity (module), 370  
 nipyapi.nifi.models.process\_groups\_entity (module), 371  
 nipyapi.nifi.models.processor\_config\_dto (module), 371  
 nipyapi.nifi.models.processor\_dto (module), 375  
 nipyapi.nifi.models.processor\_entity (module), 379  
 nipyapi.nifi.models.processor\_status\_dto (module), 381  
 nipyapi.nifi.models.processor\_status\_entity (module), 382  
 nipyapi.nifi.models.processor\_status\_snapshot\_dto (module), 383  
 nipyapi.nifi.models.processor\_status\_snapshot\_entity (module), 387  
 nipyapi.nifi.models.processor\_types\_entity (module), 388  
 nipyapi.nifi.models.processors\_entity (module), 388  
 nipyapi.nifi.models.property\_descriptor\_dto (module), 389  
 nipyapi.nifi.models.property\_descriptor\_entity (module), 392  
 nipyapi.nifi.models.property\_history\_dto (module), 392  
 nipyapi.nifi.models.provenance\_dto (module), 393  
 nipyapi.nifi.models.provenance\_entity (module), 394  
 nipyapi.nifi.models.provenance\_event\_dto (module), 395  
 nipyapi.nifi.models.provenance\_event\_entity (module), 402  
 nipyapi.nifi.models.provenance\_link\_dto (module), 402  
 nipyapi.nifi.models.provenance\_node\_dto (module), 403  
 nipyapi.nifi.models.provenance\_options\_dto (module), 405  
 nipyapi.nifi.models.provenance\_options\_entity (module), 406  
 nipyapi.nifi.models.provenance\_request\_dto (module), 406  
 nipyapi.nifi.models.provenance\_results\_dto (module), 408  
 nipyapi.nifi.models.provenance\_searchable\_field\_dto (module), 410  
 nipyapi.nifi.models.queue\_size\_dto (module), 411  
 nipyapi.nifi.models.registry\_client\_entity (module), 411  
 nipyapi.nifi.models.registry\_clients\_entity (module), 413  
 nipyapi.nifi.models.registry\_dto (module), 414  
 nipyapi.nifi.models.relationship\_dto (module), 414  
 nipyapi.nifi.models.remote\_process\_group\_contents\_dto (module), 415  
 nipyapi.nifi.models.remote\_process\_group\_dto (module), 416  
 nipyapi.nifi.models.remote\_process\_group\_entity (module), 421  
 nipyapi.nifi.models.remote\_process\_group\_port\_dto (module), 421



(*module*), 424  
nipyapi.nifi.models.remote\_process\_group\_entity (*module*), 426  
nipyapi.nifi.models.remote\_process\_group\_snapshot (*module*), 428  
nipyapi.nifi.models.remote\_process\_group\_snapshot\_entity (*module*), 430  
nipyapi.nifi.models.remote\_process\_group\_snapshot\_entity\_template\_dto (*module*), 433  
nipyapi.nifi.models.remote\_process\_group\_snapshot\_entity\_template\_entity (*module*), 433  
nipyapi.nifi.models.reporting\_task\_dto (*module*), 434  
nipyapi.nifi.models.reporting\_task\_entity (*module*), 438  
nipyapi.nifi.models.reporting\_task\_types\_entity (*module*), 440  
nipyapi.nifi.models.reporting\_tasks\_entity (*module*), 441  
nipyapi.nifi.models.resource\_dto (*module*), 441  
nipyapi.nifi.models.resources\_entity (*module*), 442  
nipyapi.nifi.models.revision\_dto (*module*), 443  
nipyapi.nifi.models.schedule\_components\_entity (*module*), 443  
nipyapi.nifi.models.search\_results\_dto (*module*), 444  
nipyapi.nifi.models.search\_results\_entity (*module*), 446  
nipyapi.nifi.models.snippet\_dto (*module*), 447  
nipyapi.nifi.models.snippet\_entity (*module*), 449  
nipyapi.nifi.models.start\_version\_control\_request\_entity (*module*), 450  
nipyapi.nifi.models.state\_entry\_dto (*module*), 451  
nipyapi.nifi.models.state\_map\_dto (*module*), 451  
nipyapi.nifi.models.status\_descriptor\_dto (*module*), 452  
nipyapi.nifi.models.status\_history\_dto (*module*), 453  
nipyapi.nifi.models.status\_history\_entity (*module*), 455  
nipyapi.nifi.models.status\_snapshot\_dto (*module*), 455  
nipyapi.nifi.models.storage\_usage\_dto (*module*), 456  
nipyapi.nifi.models.streaming\_output (*module*), 457  
nipyapi.nifi.models.submit\_replay\_request\_entity (*module*), 458  
nipyapi.nifi.models.system\_diagnostics\_dto (*module*), 459  
nipyapi.nifi.models.system\_diagnostics\_entity (*module*), 459  
nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto (*module*), 460  
nipyapi.nifi.models.system\_diagnostics\_snapshot\_entity (*module*), 465  
nipyapi.nifi.models.template\_entity (*module*), 466  
nipyapi.nifi.models.templates\_entity (*module*), 468  
nipyapi.nifi.models.tenant\_dto (*module*), 468  
nipyapi.nifi.models.tenant\_entity (*module*), 469  
nipyapi.nifi.models.tenants\_entity (*module*), 471  
nipyapi.nifi.models.transaction\_result\_entity (*module*), 471  
nipyapi.nifi.models.update\_controller\_service\_reference\_dto (*module*), 472  
nipyapi.nifi.models.user\_dto (*module*), 474  
nipyapi.nifi.models.user\_entity (*module*), 475  
nipyapi.nifi.models.user\_group\_dto (*module*), 476  
nipyapi.nifi.models.user\_group\_entity (*module*), 478  
nipyapi.nifi.models.user\_groups\_entity (*module*), 479  
nipyapi.nifi.models.users\_entity (*module*), 480  
nipyapi.nifi.models.variable\_dto (*module*), 481  
nipyapi.nifi.models.variable\_entity (*module*), 481  
nipyapi.nifi.models.variable\_registry\_dto (*module*), 482  
nipyapi.nifi.models.variable\_registry\_entity (*module*), 483  
nipyapi.nifi.models.variable\_registry\_update\_request\_entity (*module*), 484  
nipyapi.nifi.models.variable\_registry\_update\_request\_entity (*module*), 486  
nipyapi.nifi.models.variable\_registry\_update\_step\_dto (*module*), 487  
nipyapi.nifi.models.version\_control\_component\_mappings\_entity (*module*), 487  
nipyapi.nifi.models.version\_control\_information\_dto (*module*), 489

[nipyapi.nifi.models.version\\_control\\_information\\_entity \(module\)](#), 491  
[nipyapi.nifi.models.version\\_info\\_dto \(module\)](#), 491  
[nipyapi.nifi.models.versioned\\_connection \(module\)](#), 493  
[nipyapi.nifi.models.versioned\\_controller\\_service \(module\)](#), 497  
[nipyapi.nifi.models.versioned\\_flow \(module\)](#), 499  
[nipyapi.nifi.models.versioned\\_flow\\_coordinates \(module\)](#), 501  
[nipyapi.nifi.models.versioned\\_flow\\_dto \(module\)](#), 503  
[nipyapi.nifi.models.versioned\\_flow\\_entity \(module\)](#), 504  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot \(module\)](#), 504  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_entity \(module\)](#), 506  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_metadata\\_entity \(module\)](#), 507  
[nipyapi.nifi.models.versioned\\_flow\\_snapshot\\_metadata\\_set\\_entity \(module\)](#), 508  
[nipyapi.nifi.models.versioned\\_flow\\_update\\_request\\_entity \(module\)](#), 509  
[nipyapi.nifi.models.versioned\\_flow\\_update\\_request\\_entity \(module\)](#), 510  
[nipyapi.nifi.models.versioned\\_flows\\_entity \(module\)](#), 511  
[nipyapi.nifi.models.versioned\\_funnel \(module\)](#), 512  
[nipyapi.nifi.models.versioned\\_label \(module\)](#), 513  
[nipyapi.nifi.models.versioned\\_port \(module\)](#), 515  
[nipyapi.nifi.models.versioned\\_process\\_group \(module\)](#), 517  
[nipyapi.nifi.models.versioned\\_processor \(module\)](#), 521  
[nipyapi.nifi.models.versioned\\_property\\_descriptor \(module\)](#), 526  
[nipyapi.nifi.models.versioned\\_remote\\_group \(module\)](#), 527  
[nipyapi.nifi.models.versioned\\_remote\\_processor\\_group \(module\)](#), 529  
[nipyapi.nifi.rest \(module\)](#), 536  
[nipyapi.parameters \(module\)](#), 19  
[nipyapi.registry.api\\_client \(module\)](#), 619  
[nipyapi.registry.apis.access\\_api \(module\)](#), 537  
[nipyapi.registry.apis.bucket\\_flows\\_api \(module\)](#), 543  
[nipyapi.registry.apis.buckets\\_api \(module\)](#), 543  
[nipyapi.registry.apis.flows\\_api \(module\)](#), 543  
[nipyapi.registry.apis.items\\_api \(module\)](#), 543  
[nipyapi.registry.apis.policies\\_api \(module\)](#), 543  
[nipyapi.registry.apis.tenants\\_api \(module\)](#), 543  
[nipyapi.registry.configuration \(module\)](#), 543  
[nipyapi.registry.models.access\\_policy \(module\)](#), 568  
[nipyapi.registry.models.access\\_policy\\_summary \(module\)](#), 569  
[nipyapi.registry.models.batch\\_size \(module\)](#), 571  
[nipyapi.registry.models.bucket \(module\)](#), 571  
[nipyapi.registry.models.bucket\\_item \(module\)](#), 571  
[nipyapi.registry.models.bundle \(module\)](#), 571  
[nipyapi.registry.models.data\\_set\\_entity \(module\)](#), 571  
[nipyapi.registry.models.connectable\\_component \(module\)](#), 575  
[nipyapi.registry.models.controller\\_service\\_api \(module\)](#), 577  
[nipyapi.registry.models.current\\_user \(module\)](#), 577  
[nipyapi.registry.models.fields \(module\)](#), 578  
[nipyapi.registry.models.permissions \(module\)](#), 579  
[nipyapi.registry.models.resource \(module\)](#), 580  
[nipyapi.registry.models.resource\\_permissions \(module\)](#), 580  
[nipyapi.registry.models.tenant \(module\)](#), 581  
[nipyapi.registry.models.user \(module\)](#), 583  
[nipyapi.registry.models.user\\_group \(module\)](#), 584  
[nipyapi.registry.models.versioned\\_connection \(module\)](#), 585  
[nipyapi.registry.models.versioned\\_controller\\_service \(module\)](#), 589  
[nipyapi.registry.models.versioned\\_flow \(module\)](#), 592  
[nipyapi.registry.models.versioned\\_flow\\_coordinates \(module\)](#), 594  
[nipyapi.registry.models.versioned\\_flow\\_snapshot \(module\)](#), 595  
[nipyapi.registry.models.versioned\\_flow\\_snapshot\\_metadata \(module\)](#), 597

nipyapi.registry.models.versioned\_funnelnode\_snapshots (nipyapi.nifi.models.process\_group\_status\_dto.ProcessGroupStatusDTO attribute), 363  
 nipyapi.registry.models.versioned\_label\_node\_snapshots (nipyapi.nifi.models.processor\_status\_dto.ProcessorStatusDTO attribute), 382  
 nipyapi.registry.models.versioned\_port\_node\_snapshots (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO attribute), 429  
 nipyapi.registry.models.versioned\_process\_group\_snapshots (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO attribute), 454  
 nipyapi.registry.models.versioned\_processor\_node\_snapshots (nipyapi.nifi.models.system\_diagnostics\_dto.SystemDiagnosticsDTO attribute), 459  
 nipyapi.registry.models.versioned\_property\_node\_snapshot\_time (nipyapi.nifi.models.node\_dto.NodeDTO attribute), 326  
 nipyapi.registry.models.versioned\_remote\_node\_connection\_status\_snapshot\_dto (class in nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto), 323  
 nipyapi.registry.models.versioned\_remote\_process\_group\_node\_counters\_snapshot\_dto (class in nipyapi.nifi.models.node\_counters\_snapshot\_dto), 324  
 nipyapi.registry.rest (module), 621  
 nipyapi.security (module), 21  
 nipyapi.system (module), 25  
 nipyapi.templates (module), 26  
 nipyapi.utils (module), 28  
 nipyapi.versioning (module), 31  
 node (nipyapi.nifi.models.node\_entity.NodeEntity attribute), 326  
 node\_address (nipyapi.nifi.models.bulletin\_dto.BulletinDTO attribute), 220  
 node\_address (nipyapi.nifi.models.bulletin\_entity.BulletinEntity attribute), 221  
 node\_id (nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto.NodeConnectionStatusSnapshotDTO attribute), 323  
 node\_id (nipyapi.nifi.models.node\_counters\_snapshot\_dto.NodeCountersSnapshotDTO attribute), 324  
 node\_id (nipyapi.nifi.models.node\_dto.NodeDTO attribute), 325  
 node\_id (nipyapi.nifi.models.node\_port\_status\_snapshot\_dto.NodePortStatusSnapshotDTO attribute), 328  
 node\_id (nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto.NodeProcessGroupStatusSnapshotDTO attribute), 329  
 node\_id (nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.NodeProcessorStatusSnapshotDTO attribute), 330  
 node\_id (nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto.NodeRemoteProcessGroupStatusSnapshotDTO attribute), 331  
 node\_id (nipyapi.nifi.models.node\_status\_snapshots\_dto.NodeStatusSnapshotsDTO attribute), 333  
 node\_id (nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto.NodeSystemDiagnosticsSnapshotDTO attribute), 334  
 node\_results (nipyapi.nifi.models.cluster\_search\_results\_entity.ClusterSearchResultsEntity attribute), 225  
 node\_snapshots (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO attribute), 245  
 node\_snapshots (nipyapi.nifi.models.counters\_dto.CountersDTO attribute), 280  
 node\_snapshots (nipyapi.nifi.models.port\_status\_dto.PortStatusDTO attribute), 342  
 object\_count (nipyapi.nifi.models.queue\_size\_dto.QueueSizeDTO attribute), 333



attribute), 411  
 oidc\_callback() (nipyapi.registry.apis.access\_api.AccessApi  
   method), 540  
 oidc\_callback\_with\_http\_info()  
   (nipyapi.registry.apis.access\_api.AccessApi  
   method), 541  
 oidc\_exchange() (nipyapi.registry.apis.access\_api.AccessApi  
   method), 541  
 oidc\_exchange\_with\_http\_info()  
   (nipyapi.registry.apis.access\_api.AccessApi  
   method), 541  
 oidc\_logout() (nipyapi.registry.apis.access\_api.AccessApi  
   method), 541  
 oidc\_logout\_with\_http\_info()  
   (nipyapi.registry.apis.access\_api.AccessApi  
   method), 542  
 oidc\_request() (nipyapi.registry.apis.access\_api.AccessApi  
   method), 542  
 oidc\_request\_with\_http\_info()  
   (nipyapi.registry.apis.access\_api.AccessApi  
   method), 542  
 oidclogin\_supported  
   (nipyapi.registry.models.current\_user.CurrentUser  
   attribute), 578  
 oldest\_event (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO  
   attribute), 409  
 operate\_permissions  
   (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity  
   attribute), 266  
 operate\_permissions  
   (nipyapi.nifi.models.controller\_service\_referencing\_component\_entity.ControllerServiceReferencingComponentEntity  
   attribute), 270  
 operate\_permissions  
   (nipyapi.nifi.models.port\_entity.PortEntity  
   attribute), 341  
 operate\_permissions  
   (nipyapi.nifi.models.processor\_entity.ProcessorEntity  
   attribute), 380  
 operate\_permissions  
   (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity  
   attribute), 423  
 operate\_permissions  
   (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity  
   attribute), 427  
 operate\_permissions  
   (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity  
   attribute), 439  
 operation (nipyapi.nifi.models.action\_dto.ActionDTO  
   attribute), 203  
 OPTIONS() (nipyapi.nifi.rest.RESTClientObject  
   method), 536  
 OPTIONS() (nipyapi.registry.rest.RESTClientObject  
   method), 622  
 origin\_x (nipyapi.nifi.models.copy\_snippet\_request\_entity.CopySnippetRequestEntity  
   attribute), 277  
 origin\_y (nipyapi.nifi.models.copy\_snippet\_request\_entity.CopySnippetRequestEntity  
   attribute), 277  
 origin\_y (nipyapi.nifi.models.instantiate\_template\_request\_entity.InstantiateTemplateRequestEntity  
   attribute), 312  
 original (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO  
   attribute), 290  
 original\_count (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO  
   attribute), 290  
 original\_size (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO  
   attribute), 290  
 os\_architecture (nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO  
   attribute), 493  
 os\_name (nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO  
   attribute), 493  
 os\_version (nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO  
   attribute), 493  
 output (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO  
   attribute), 250  
 output (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO  
   attribute), 345  
 output (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO  
   attribute), 400  
 output (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO  
   attribute), 386  
 output\_available  
   (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO  
   attribute), 400  
 output\_content\_claim\_file\_size  
   (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO  
   attribute), 400  
 output\_content\_claim\_file\_size\_bytes  
   (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO  
   attribute), 400  
 output\_content\_claim\_section  
   (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO  
   attribute), 400  
 output\_port\_count  
   (nipyapi.nifi.models.controller\_dto.ControllerDTO  
   attribute), 257  
 output\_port\_count  
   (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO  
   attribute), 423

output\_port\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 358  
 output\_port\_count (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 419  
 output\_port\_count (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity attribute), 423  
 output\_port\_results (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO attribute), 446  
 output\_port\_status\_snapshots (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 368  
 output\_ports (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 257  
 output\_ports (nipyapi.nifi.models.flow\_dto.FlowDTO attribute), 297  
 output\_ports (nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO attribute), 304  
 output\_ports (nipyapi.nifi.models.output\_ports\_entity.OutputPortsEntity attribute), 334  
 output\_ports (nipyapi.nifi.models.remote\_process\_group\_contents\_dto.RemoteProcessGroupContentsDTO attribute), 416  
 output\_ports (nipyapi.nifi.models.snippet\_dto.SnippetDTO attribute), 448  
 output\_ports (nipyapi.nifi.models.versioned\_process\_group\_entity.VersionedProcessGroupEntity attribute), 520  
 output\_ports (nipyapi.nifi.models.versioned\_remote\_process\_group\_entity.VersionedRemoteProcessGroupEntity attribute), 531  
 output\_ports (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606  
 output\_ports (nipyapi.registry.models.versioned\_remote\_process\_group\_entity.VersionedRemoteProcessGroupEntity attribute), 617  
 OutputPortsApi (class in nipyapi.nifi.apis.output\_ports\_api), 102  
 OutputPortsEntity (class in nipyapi.nifi.models.output\_ports\_entity), 334  
 parameter\_context (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 353  
 parameter\_context (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 358  
 parameter\_context (nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO attribute), 361  
 parameter\_context\_name (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroup attribute), 520  
 parameter\_context\_name (nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroup attribute), 606  
 parameter\_context\_permissions (nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity attribute), 285  
 parameter\_context\_results (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO attribute), 446  
 parameter\_contexts (nipyapi.nifi.models.versioned\_flow\_snapshot.VersionedFlowSnapshot attribute), 506  
 parameter\_contexts (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 257  
 parameter\_contexts (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO attribute), 446  
 ParameterContextsApi (class in nipyapi.nifi.apis.parameter\_contexts\_api), 105  
 ParametersEntity.to\_tuples () (nipyapi.nifi.api\_client.ApiClient method), 634  
 ParametersEntity.to\_tuples () (nipyapi.registry.api\_client.ApiClient method), 620  
 parent\_group (nipyapi.nifi.models.component\_search\_result\_dto.ComponentSearchResultDTO attribute), 234  
 parent\_group\_id (nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO attribute), 198  
 parent\_group\_id (nipyapi.nifi.models.component\_reference\_dto.ComponentReferenceDTO attribute), 230  
 parent\_group\_id (nipyapi.nifi.models.component\_reference\_entity.ComponentReferenceEntity attribute), 232  
 parent\_group\_id (nipyapi.nifi.models.connection\_dto.ConnectionDTO attribute), 241  
 parent\_group\_id (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO attribute), 263  
 parent\_group\_id (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity attribute), 266  
 parent\_group\_id (nipyapi.nifi.models.funnel\_dto.FunnelDTO attribute), 305  
 parent\_group\_id (nipyapi.nifi.models.label\_dto.LabelDTO attribute), 313  
 parent\_group\_id (nipyapi.nifi.models.port\_dto.PortDTO attribute), 339  
 parent\_group\_id (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 353

## P

parent\_group\_id (nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO attribute), 361  
 parent\_group\_id (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 317  
 parent\_group\_id (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 377  
 parent\_group\_id (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO attribute), 419  
 parent\_group\_id (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO attribute), 437  
 parent\_group\_id (nipyapi.nifi.models.snippet\_dto.SnippetDTO attribute), 394  
 parent\_group\_id (nipyapi.nifi.models.snippet\_dto.SnippetDTO attribute), 448  
 parent\_group\_id (nipyapi.nifi.models.tenant\_dto.TenantDTO attribute), 469  
 parent\_group\_id (nipyapi.nifi.models.user\_dto.UserDTO attribute), 474  
 parent\_group\_id (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO attribute), 509  
 parent\_group\_id (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO attribute), 477  
 parent\_uuids (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 401  
 parent\_uuids (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO attribute), 404  
 partitioning\_attribute (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 496  
 partitioning\_attribute (nipyapi.nifi.models.versioned\_connection.VersionedConnection attribute), 496  
 partitioning\_attribute (nipyapi.registry.models.versioned\_connection.VersionedConnection attribute), 588  
 PATCH () (nipyapi.nifi.rest.RESTClientObject method), 536  
 PATCH () (nipyapi.registry.rest.RESTClientObject method), 622  
 PeerDTO (class in nipyapi.nifi.models.peer\_dto), 335  
 peers (nipyapi.nifi.models.peers\_entity.PeersEntity attribute), 336  
 PeersEntity (class in nipyapi.nifi.models.peers\_entity), 336  
 penalized (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 300  
 penalized (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 303  
 penalty\_duration (nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO attribute), 374  
 penalty\_duration (nipyapi.nifi.models.versioned\_processor\_versioned\_processor attribute), 524  
 penalty\_duration (nipyapi.registry.models.versioned\_processor\_versioned\_processor attribute), 610  
 penalty\_expires\_in (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 300  
 penalty\_expires\_in (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 303  
 percent\_completed (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO attribute), 290

[attribute](#)), 358  
[permissions \(nipyapi.nifi.models.process\\_group\\_flow\\_entity.ProcessGroupFlowEntity \(class in nipyapi.nifi.models.process\\_group\\_flow\\_entity\), 340](#)  
[attribute](#)), 362  
[permissions \(nipyapi.nifi.models.processor\\_entity.ProcessorEntity \(class in nipyapi.nifi.models.processor\\_entity\), 340](#)  
[attribute](#)), 380  
[permissions \(nipyapi.nifi.models.registry\\_client\\_entity.RegistryClientEntity \(class in nipyapi.nifi.models.registry\\_client\\_entity\), 342](#)  
[attribute](#)), 412  
[permissions \(nipyapi.nifi.models.remote\\_process\\_group\\_entity.RemoteProcessGroupEntity \(class in nipyapi.nifi.models.remote\\_process\\_group\\_entity\), 343](#)  
[attribute](#)), 423  
[permissions \(nipyapi.nifi.models.remote\\_process\\_group\\_port\\_entity.RemoteProcessGroupPortEntity \(class in nipyapi.nifi.models.remote\\_process\\_group\\_port\\_entity\), 344](#)  
[attribute](#)), 427  
[permissions \(nipyapi.nifi.models.reporting\\_task\\_entity.ReportingTaskEntity \(class in nipyapi.nifi.models.reporting\\_task\\_entity\), 344](#)  
[attribute](#)), 440  
[permissions \(nipyapi.nifi.models.template\\_entity.TemplateEntity \(class in nipyapi.nifi.models.template\\_entity\), 346](#)  
[attribute](#)), 467  
[permissions \(nipyapi.nifi.models.tenant\\_entity.TenantEntity \(class in nipyapi.nifi.models.tenant\\_entity\), 195](#)  
[attribute](#)), 470  
[permissions \(nipyapi.nifi.models.user\\_entity.UserEntity \(class in nipyapi.nifi.models.user\\_entity\), 197](#)  
[attribute](#)), 476  
[permissions \(nipyapi.nifi.models.user\\_group\\_entity.UserGroupEntity \(class in nipyapi.nifi.models.user\\_group\\_entity\), 198](#)  
[attribute](#)), 479  
[permissions \(nipyapi.nifi.models.versioned\\_flow.VersionedFlow \(class in nipyapi.nifi.models.versioned\\_flow\), 200](#)  
[attribute](#)), 501  
[permissions \(nipyapi.registry.models.bucket.Bucket \(class in nipyapi.registry.models.bucket\), 208](#)  
[attribute](#)), 572  
[permissions \(nipyapi.registry.models.bucket\\_item.BucketItem \(class in nipyapi.registry.models.bucket\\_item\), 231](#)  
[attribute](#)), 574  
[permissions \(nipyapi.registry.models.versioned\\_flow.VersionedFlow \(class in nipyapi.registry.models.versioned\\_flow\), 232](#)  
[attribute](#)), 593  
[PermissionsDTO \(class in nipyapi.nifi.models.permissions\\_dto\), 337](#)  
[persists\\_state \(nipyapi.nifi.models.controller\\_service\\_dto.ControllerServiceDTO \(class in nipyapi.nifi.models.controller\\_service\\_dto\), 263](#)  
[attribute](#)), 263  
[persists\\_state \(nipyapi.nifi.models.processor\\_dto.ProcessorDTO \(class in nipyapi.nifi.models.processor\\_dto\), 266](#)  
[attribute](#)), 377  
[persists\\_state \(nipyapi.nifi.models.reporting\\_task\\_dto.ReportingTaskDTO \(class in nipyapi.nifi.models.reporting\\_task\\_dto\), 266](#)  
[attribute](#)), 437  
[policies \(nipyapi.registry.models.resource\\_permissions.ResourcePermissions \(class in nipyapi.registry.models.resource\\_permissions\), 271](#)  
[attribute](#)), 581  
[policies\\_permissions \(nipyapi.nifi.models.current\\_user\\_entity.CurrentUserEntity \(class in nipyapi.nifi.models.current\\_user\\_entity\), 285](#)  
[PoliciesApi \(class in nipyapi.nifi.apis.policies\\_api\), 113](#)  
[PoliciesApi \(class in nipyapi.registry.apis.policies\\_api\), 559](#)  
[port \(nipyapi.nifi.models.peer\\_dto.PeerDTO \(class in nipyapi.nifi.models.peer\\_dto\), 335](#)  
[port\\_status \(nipyapi.nifi.models.port\\_status\\_entity.PortStatusEntity \(class in nipyapi.nifi.models.port\\_status\\_entity\), 343](#)  
[port\\_status\\_snapshot \(nipyapi.nifi.models.port\\_status\\_snapshot\\_entity.PortStatusSnapshotEntity \(class in nipyapi.nifi.models.port\\_status\\_snapshot\\_entity\), 346](#)  
[port\\_type \(nipyapi.nifi.models.port\\_entity.PortEntity \(class in nipyapi.nifi.models.port\\_entity\), 341](#)



position (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO (class in  
attribute), 353 nipyapi.registry.models.versioned\_controller\_service.VersionedControllerServiceEntity (class in  
attribute), 591 nipyapi.registry.models.versioned\_funnel.VersionedFunnelEntity (class in  
attribute), 599 nipyapi.registry.models.versioned\_label.VersionedLabelEntity (class in  
attribute), 601 nipyapi.registry.models.versioned\_port.VersionedPortEntity (class in  
attribute), 603 nipyapi.registry.models.versioned\_process\_group.VersionedProcessGroupEntity (class in  
attribute), 606 nipyapi.registry.models.versioned\_processor.VersionedProcessorEntity (class in  
attribute), 610 nipyapi.registry.models.versioned\_remote\_group\_port.VersionedRemoteGroupPortEntity (class in  
attribute), 614 nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroupEntity (class in  
attribute), 617 nipyapi.registry.rest.RESTClientObject (class in  
attribute), 622 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity (class in  
attribute), 599 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 606 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.processor\_dto.ProcessorDTO (class in  
attribute), 377 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 601 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.processor\_entity.ProcessorEntity (class in  
attribute), 380 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 603 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity (class in  
attribute), 412 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 606 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO (class in  
attribute), 419 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 610 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity (class in  
attribute), 423 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 614 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity (class in  
attribute), 427 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 617 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO (class in  
attribute), 437 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity (class in  
attribute), 440 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 536 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 622 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.template\_entity.TemplateEntity (class in  
attribute), 467 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 622 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.tenant\_dto.TenantDTO (class in  
attribute), 469 nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO (class in  
attribute), 250 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.tenant\_entity.TenantEntity (class in  
attribute), 470 nipyapi.parameters), 20  
position (nipyapi.nifi.models.user\_dto.UserDTO (class in  
attribute), 474 nipyapi.nifi.api\_client.ApiClient (class in  
attribute), 534 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 622 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.user\_entity.UserEntity (class in  
attribute), 476 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 622 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO (class in  
attribute), 477 nipyapi.nifi.rest.RESTClientObject (class in  
attribute), 620 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity (class in  
attribute), 479 nipyapi.nifi.models.attribute\_dto.AttributeDTO (class in  
attribute), 211 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_connection.VersionedConnectionEntity (class in  
attribute), 496 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerServiceEntity (class in  
attribute), 499 nipyapi.nifi.models.property\_history\_dto.PropertyHistoryDTO (class in  
attribute), 393 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_funnel.VersionedFunnelEntity (class in  
attribute), 513 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_label.VersionedLabelEntity (class in  
attribute), 514 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_port.VersionedPortEntity (class in  
attribute), 516 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_process\_group.VersionedProcessGroupEntity (class in  
attribute), 520 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_processor.VersionedProcessorEntity (class in  
attribute), 524 nipyapi.nifi.models.prioritizer\_types\_entity.PrioritizerTypesEntity (class in  
attribute), 348 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_remote\_group\_port\_entity.VersionedRemoteGroupPortEntity (class in  
attribute), 528 nipyapi.nifi.models.connection\_dto.ConnectionDTO (class in  
attribute), 241 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.nifi.models.versioned\_remote\_process\_group\_entity.VersionedRemoteProcessGroupEntity (class in  
attribute), 531 nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO (class in  
attribute), 496 nipyapi.nifi.models.position\_dto), 347  
position (nipyapi.registry.models.versioned\_connection.VersionedConnectionEntity (class in  
attribute), 588 nipyapi.registry.models.versioned\_connection.VersionedConnectionEntity (class in  
attribute), 588 nipyapi.registry.models.versioned\_connection.VersionedConnectionEntity (class in  
attribute), 588

PrioritizerTypesEntity (class in attribute), 368  
 (nipyapi.nifi.models.prioritizer\_types\_entity), process\_groups (nipyapi.nifi.models.flow\_dto.FlowDTO  
 attribute), 297  
 process\_group (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity), 305  
 (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity), 305  
 process\_group\_flow process\_groups (nipyapi.nifi.models.process\_groups\_entity.ProcessGroupsEntity), 362  
 (nipyapi.nifi.models.process\_group\_flow\_entity.ProcessGroupFlowEntity), 362  
 process\_group\_id (nipyapi.nifi.models.affected\_component\_dto.AffectedComponentDTO), 206  
 (nipyapi.nifi.models.affected\_component\_dto.AffectedComponentDTO), 206  
 process\_group\_id (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO), 228  
 (nipyapi.nifi.models.component\_difference\_dto.ComponentDifferenceDTO), 228  
 process\_group\_id (nipyapi.nifi.models.create\_active\_request\_entity.CreateActiveRequestEntity), 282  
 (nipyapi.nifi.models.create\_active\_request\_entity.CreateActiveRequestEntity), 282  
 process\_group\_id (nipyapi.nifi.models.variable\_dto.VariableDTO), 481  
 (nipyapi.nifi.models.variable\_dto.VariableDTO), 481  
 process\_group\_id (nipyapi.nifi.models.variable\_registry\_dto.VariableRegistryDTO), 482  
 (nipyapi.nifi.models.variable\_registry\_dto.VariableRegistryDTO), 482  
 process\_group\_id (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO), 485  
 (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO), 485  
 process\_group\_id (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO), 509  
 (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO), 509  
 process\_group\_results (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO), 446  
 (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO), 446  
 process\_group\_revision ProcessGroupsApi (class in nipyapi.nifi.apis.process\_groups\_api), 116  
 (nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity), 450  
 (nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity), 450  
 process\_group\_revision (nipyapi.nifi.models.variable\_registry\_entity.VariableRegistryEntity), 483  
 (nipyapi.nifi.models.variable\_registry\_entity.VariableRegistryEntity), 483  
 process\_group\_revision (nipyapi.nifi.models.variable\_registry\_update\_request\_entity.VariableRegistryUpdateRequestEntity), 486  
 (nipyapi.nifi.models.variable\_registry\_update\_request\_entity.VariableRegistryUpdateRequestEntity), 486  
 process\_group\_revision (nipyapi.nifi.models.version\_control\_component\_mappings\_entity.VersionControlComponentMappingsEntity), 488  
 (nipyapi.nifi.models.version\_control\_component\_mappings\_entity.VersionControlComponentMappingsEntity), 488  
 process\_group\_revision (nipyapi.nifi.models.version\_control\_information\_entity.VersionControlInformationEntity), 491  
 (nipyapi.nifi.models.version\_control\_information\_entity.VersionControlInformationEntity), 491  
 process\_group\_revision (nipyapi.nifi.models.versioned\_flow\_snapshot\_entity.VersionedFlowSnapshotEntity), 507  
 (nipyapi.nifi.models.versioned\_flow\_snapshot\_entity.VersionedFlowSnapshotEntity), 507  
 process\_group\_revision (nipyapi.nifi.models.versioned\_flow\_update\_request\_entity.VersionedFlowUpdateRequestEntity), 511  
 (nipyapi.nifi.models.versioned\_flow\_update\_request\_entity.VersionedFlowUpdateRequestEntity), 511  
 process\_group\_status (nipyapi.nifi.models.process\_group\_status\_entity.ProcessGroupStatusEntity), 364  
 (nipyapi.nifi.models.process\_group\_status\_entity.ProcessGroupStatusEntity), 364  
 process\_group\_status\_snapshot processor\_status\_snapshot (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity), 371  
 (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity), 371  
 process\_group\_status\_snapshots processor\_status\_snapshots (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity), 387  
 (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity), 387

713

ProvenanceEventEntity (class in attribute), 358  
 (nipyapi.nifi.models.provenance\_event\_entity), public\_output\_port\_count  
 402 (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO  
 attribute), 353

ProvenanceEventsApi (class in attribute), 353  
 (nipyapi.nifi.apis.provenance\_events\_api), public\_output\_port\_count  
 153 (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity  
 attribute), 359

ProvenanceLinkDTO (class in attribute), 359  
 (nipyapi.nifi.models.provenance\_link\_dto), purge\_connection() (in module nipyapi.canvas),  
 402 14

ProvenanceNodeDTO (class in purge\_process\_group() (in module  
 (nipyapi.nifi.models.provenance\_node\_dto), nipyapi.canvas), 14  
 403 PUT() (nipyapi.nifi.rest.RESTClientObject method), 536

ProvenanceOptionsDTO (class in PUT() (nipyapi.registry.rest.RESTClientObject method),  
 (nipyapi.nifi.models.provenance\_options\_dto), 622  
 405

ProvenanceOptionsEntity (class in **Q**  
 (nipyapi.nifi.models.provenance\_options\_entity), query\_history() (nipyapi.nifi.apis.flow\_api.FlowApi  
 406 method), 88

ProvenanceRequestDTO (class in query\_history\_with\_http\_info()  
 (nipyapi.nifi.models.provenance\_request\_dto), (nipyapi.nifi.apis.flow\_api.FlowApi method),  
 406 89

ProvenanceResultsDTO (class in queue\_size(nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO  
 (nipyapi.nifi.models.provenance\_results\_dto), attribute), 321  
 408 queued(nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionS  
 attribute), 250

ProvenanceSearchableFieldDTO (class in attribute), 250  
 (nipyapi.nifi.models.provenance\_searchable\_field\_dto), queued(nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO  
 410 attribute), 275

proxy (nipyapi.registry.models.resource\_permissions.ResourcePermissions (nipyapi.nifi.models.node\_dto.NodeDTO  
 attribute), 581 attribute), 326

proxy\_host (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO process\_group\_status\_snapshot\_dto.ProcessG  
 attribute), 419 attribute), 368

proxy\_host (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup connection\_status\_snapshot\_dto.Conn  
 attribute), 532 attribute), 250

proxy\_host (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup process\_group\_status\_snapshot\_dto.P  
 attribute), 618 attribute), 369

proxy\_password (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO models.flow\_file\_dto.FlowFileDTO  
 attribute), 420 attribute), 300

proxy\_port (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFil  
 attribute), 420 attribute), 303

proxy\_port (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup connection\_status\_snapshot\_dto.Conn  
 attribute), 532 attribute), 250

proxy\_port (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup process\_group\_status\_snapshot\_dto.P  
 attribute), 618 attribute), 369

proxy\_user (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO (class in  
 attribute), 420 nipyapi.nifi.models.queue\_size\_dto), 411

proxy\_user (nipyapi.nifi.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup  
 attribute), 532

**R**

proxy\_user (nipyapi.registry.models.versioned\_remote\_process\_group.VersionedRemoteProcessGroup  
 attribute), 618 read(nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGro  
 attribute), 369

public\_input\_port\_count read(nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusS  
 (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 386  
 attribute), 353

public\_input\_port\_count receive\_flow\_files()  
 (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity method), 64



receive\_flow\_files\_with\_http\_info() (class in remote\_site\_http\_listening\_port (nipyapi.nifi.apis.data\_transfer\_api.DataTransferApi), 65  
 received (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO (class in nipyapi.nifi.models.versioned\_remote\_group\_port\_entity.RemoteProcessGroupPortEntity), 369  
 received (nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_dto.RemoteProcessGroupStatusSnapshotDTO (class in nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity), 432  
 recurse\_flow() (in module nipyapi.canvas), 10  
 reference\_cycle (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO (class in nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO), 268  
 reference\_type (nipyapi.nifi.models.affected\_component\_dto.AffectedComponentDTO (class in nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity), 206  
 reference\_type (nipyapi.nifi.models.affected\_component\_entity.AffectedComponentEntity (class in nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity), 208  
 reference\_type (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO (class in nipyapi.nifi.models.remote\_process\_group\_status\_snapshot\_entity.RemoteProcessGroupStatusSnapshotEntity), 268  
 referencing\_component\_revisions (nipyapi.nifi.models.update\_controller\_service\_reference\_request\_entity.UpdateControllerServiceReferenceRequestEntity (class in nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO), 473  
 referencing\_components (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO (class in nipyapi.nifi.models.flow\_dto.FlowDTO), 263  
 referencing\_components (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO (class in nipyapi.nifi.models.flow\_snippet\_dto.FlowSnippetDTO), 268  
 registries (nipyapi.nifi.models.registry\_clients\_entity.RegistryClientsEntity (class in nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity), 413  
 registry\_id (nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO (class in nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity), 490  
 registry\_id (nipyapi.nifi.models.versioned\_flow\_dto.VersionedFlowDTO (class in nipyapi.nifi.models.snippet\_dto.SnippetDTO), 504  
 registry\_id (nipyapi.nifi.models.versioned\_flow\_snapshot\_entity.VersionedFlowSnapshotEntity (class in nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity), 507  
 registry\_id (nipyapi.nifi.models.versioned\_flow\_snapshot\_metadata\_entity.VersionedFlowSnapshotMetadataEntity (class in nipyapi.nifi.models.versioned\_process\_group\_entity.VersionedProcessGroupEntity), 507  
 registry\_name (nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO (class in nipyapi.nifi.models.versioned\_process\_group\_entity.VersionedProcessGroupEntity), 490  
 registry\_url (nipyapi.nifi.models.versioned\_flow\_coordinates\_entity.VersionedFlowCoordinatesEntity (class in nipyapi.nifi.models.remote\_site\_http\_listening\_port\_entity.RemoteSiteHttpListeningPortEntity), 502  
 registry\_url (nipyapi.registry.models.versioned\_flow\_coordinates\_entity.VersionedFlowCoordinatesEntity (class in nipyapi.nifi.models.remote\_site\_http\_listening\_port\_entity.RemoteSiteHttpListeningPortEntity), 595  
 RegistryClientEntity (class in nipyapi.nifi.models.registry\_client\_entity), 411  
 RegistryClientsEntity (class in nipyapi.nifi.models.registry\_clients\_entity), 413  
 RegistryDTO (class in nipyapi.nifi.models.registry\_dto), 414  
 relationship (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO (class in nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity), 401  
 RelationshipDTO (class in nipyapi.nifi.models.relationship\_dto), 415  
 relationships (nipyapi.nifi.models.processor\_dto.ProcessorDTO (class in nipyapi.nifi.models.remote\_process\_group\_port\_dto.RemoteProcessGroupPortDTO), 421

```

nipyapi.nifi.models.remote_process_group_port_dto), method), 135
424 remove_funnel() (nipyapi.nifi.apis.funnel_api.FunnelApi
RemoteProcessGroupPortEntity (class in method), 96
nipyapi.nifi.models.remote_process_group_port_entity),
426 remove_funnel_with_http_info()
(nipyapi.nifi.apis.funnel_api.FunnelApi
RemoteProcessGroupsApi (class in method), 97
nipyapi.nifi.apis.remote_process_groups_api), remove_input_port()
155 (nipyapi.nifi.apis.input_ports_api.InputPortsApi
RemoteProcessGroupsEntity (class in method), 98
nipyapi.nifi.models.remote_process_groups_entity), remove_input_port_with_http_info()
434 (nipyapi.nifi.apis.input_ports_api.InputPortsApi
RemoteProcessGroupStatusDTO (class in method), 99
nipyapi.nifi.models.remote_process_group_status_dto), remove_label() (nipyapi.nifi.apis.labels_api.LabelsApi
428 method), 101
RemoteProcessGroupStatusEntity (class in remove_label_with_http_info()
nipyapi.nifi.models.remote_process_group_status_entity), (nipyapi.nifi.apis.labels_api.LabelsApi
430 method), 101
RemoteProcessGroupStatusSnapshotDTO remove_output_port()
(class in nipyapi.nifi.models.remote_process_group_status_snapshot_dto),
430 (nipyapi.nifi.apis.output_ports_api.OutputPortsApi
method), 103
RemoteProcessGroupStatusSnapshotEntity remove_output_port_with_http_info()
(class in nipyapi.nifi.models.remote_process_group_status_snapshot_entity),
433 (nipyapi.nifi.apis.output_ports_api.OutputPortsApi
method), 103
remove_access_policy() remove_process_group()
(nipyapi.nifi.apis.policies_api.PoliciesApi (nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 114 method), 135
remove_access_policy() remove_process_group_with_http_info()
(nipyapi.registry.apis.policies_api.PoliciesApi (nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 561 method), 135
remove_access_policy_with_http_info() remove_remote_process_group()
(nipyapi.nifi.apis.policies_api.PoliciesApi (nipyapi.nifi.apis.remote_process_groups_api.RemoteProcessGro
method), 115 method), 156
remove_access_policy_with_http_info() remove_remote_process_group_with_http_info()
(nipyapi.registry.apis.policies_api.PoliciesApi (nipyapi.nifi.apis.remote_process_groups_api.RemoteProcessGro
method), 561 method), 157
remove_context_from_process_group() (in remove_reporting_task()
module nipyapi.parameters), 21 (nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi
method), 166
remove_controller_service() remove_reporting_task_with_http_info()
(nipyapi.nifi.apis.controller_services_api.ControllerServicesApi (nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi
method), 56 method), 166
remove_controller_service_with_http_info() remove_service_user() (in module
(nipyapi.nifi.apis.controller_services_api.ControllerServicesApi nipyapi.security), 25
method), 56 remove_service_user_group() (in module
nipyapi.security), 25
remove_drop_request() remove_template()
(nipyapi.nifi.apis.flowfile_queues_api.FlowfileQueuesApi (nipyapi.nifi.apis.templates_api.TemplatesApi
method), 95 method), 174
remove_drop_request() (nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi
method), 134 remove_template_with_http_info()
(nipyapi.nifi.apis.templates_api.TemplatesApi
method), 175
remove_drop_request_with_http_info() remove_user() (nipyapi.nifi.apis.tenants_api.TenantsApi
method), 95 remove_user() (nipyapi.registry.apis.tenants_api.TenantsApi
method), 178
remove_drop_request_with_http_info() remove_user() (nipyapi.registry.apis.tenants_api.TenantsApi
method), 178

```

method), 565  
 remove\_user\_group() (nipyapi.nifi.apis.tenants\_api.TenantsApi method), 178  
 remove\_user\_group() (nipyapi.registry.apis.tenants\_api.TenantsApi method), 566  
 remove\_user\_group\_with\_http\_info() (nipyapi.nifi.apis.tenants\_api.TenantsApi method), 178  
 remove\_user\_group\_with\_http\_info() (nipyapi.registry.apis.tenants\_api.TenantsApi method), 566  
 remove\_user\_with\_http\_info() (nipyapi.nifi.apis.tenants\_api.TenantsApi method), 179  
 remove\_user\_with\_http\_info() (nipyapi.registry.apis.tenants\_api.TenantsApi method), 566  
 replace\_process\_group() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 136  
 replace\_process\_group\_with\_http\_info() (nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi method), 136  
 replay\_available (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 401  
 replay\_explanation (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 401  
 reporting\_task\_bulletins (nipyapi.nifi.models.controller\_bulletins\_entity.ControllerBulletinsEntity attribute), 253  
 reporting\_task\_types (nipyapi.nifi.models.reporting\_task\_types\_entity.ReportingTaskTypesEntity attribute), 441  
 reporting\_tasks (nipyapi.nifi.models.reporting\_tasks\_entity.ReportingTasksEntity attribute), 441  
 ReportingTaskDTO (class in nipyapi.nifi.models.reporting\_task\_dto), 434  
 ReportingTaskEntity (class in nipyapi.nifi.models.reporting\_task\_entity), 439  
 ReportingTasksApi (class in nipyapi.nifi.apis.reporting\_tasks\_api), 162  
 ReportingTasksEntity (class in nipyapi.nifi.models.reporting\_tasks\_entity), 441  
 ReportingTaskTypesEntity (class in nipyapi.nifi.models.reporting\_task\_types\_entity), 440  
 request (nipyapi.nifi.models.lineage\_dto.LineageDTO attribute), 317  
 request (nipyapi.nifi.models.provenance\_dto.ProvenanceDTO attribute), 394  
 request (nipyapi.nifi.models.variable\_registry\_update\_request\_entity.VariableRegistryUpdateRequestEntity attribute), 486  
 request (nipyapi.nifi.models.versioned\_flow\_update\_request\_entity.VersionedFlowUpdateRequestEntity attribute), 511  
 request() (nipyapi.nifi.api\_client.ApiClient method), 534  
 request() (nipyapi.nifi.rest.RESTClientObject method), 536  
 request() (nipyapi.registry.api\_client.ApiClient method), 620  
 request() (nipyapi.registry.rest.RESTClientObject method), 622  
 request\_id (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO attribute), 485  
 request\_id (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO attribute), 510  
 required (nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDTO attribute), 391  
 Resource (class in nipyapi.registry.models.resource), 580  
 resource (nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO attribute), 195  
 resource (nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO attribute), 198  
 resource (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity attribute), 569  
 resource (nipyapi.registry.models.access\_policy\_summary.AccessPolicySummaryDTO attribute), 570  
 resource\_definition (nipyapi.nifi.models.versioned\_property\_descriptor.VersionedPropertyDescriptor attribute), 526  
 resource\_definition (nipyapi.registry.models.versioned\_property\_descriptor.VersionedPropertyDescriptor attribute), 526  
 resource\_permissions (nipyapi.nifi.models.current\_user.CurrentUser attribute), 578  
 resource\_permissions (nipyapi.registry.models.tenant.Tenant attribute), 582  
 resource\_permissions (nipyapi.registry.models.user.User attribute), 583  
 resource\_permissions (nipyapi.registry.models.user\_group.UserGroup attribute), 585  
 ResourceDTO (class in nipyapi.nifi.models.resource\_dto), 441  
 ResourcePermissions (class in nipyapi.registry.models.resource\_permissions), 580  
 resources (nipyapi.nifi.models.resources\_entity.ResourcesEntity attribute), 442

ResourcesApi (class in `nipyapi.nifi.apis.resources_api`), 169

ResourcesEntity (class in `nipyapi.nifi.models.resources_entity`), 442

response\_code (`nipyapi.nifi.models.transaction_result_entity.TransactionResultEntity` attribute), 472

RESTClientObject (class in `nipyapi.nifi.rest`), 536

RESTClientObject (class in `nipyapi.registry.rest`), 622

RESTResponse (class in `nipyapi.nifi.rest`), 536

RESTResponse (class in `nipyapi.registry.rest`), 622

restricted (`nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO` attribute), 263

restricted (`nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO` attribute), 288

restricted (`nipyapi.nifi.models.processor_dto.ProcessorDTO` attribute), 378

restricted (`nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO` attribute), 437

restricted\_components\_permissions (`nipyapi.nifi.models.current_user_entity.CurrentUserEntity` attribute), 285

results (`nipyapi.nifi.models.lineage_dto.LineageDTO` attribute), 317

results (`nipyapi.nifi.models.provenance_dto.ProvenanceDTO` attribute), 394

retried\_relationships (`nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO` attribute), 374

retried\_relationships (`nipyapi.nifi.models.versioned_processor.VersionedProcessor` attribute), 524

retried\_relationships (`nipyapi.registry.models.versioned_processor.VersionedProcessor` attribute), 610

retry (`nipyapi.nifi.models.relationship_dto.RelationshipDTO` attribute), 415

retry\_count (`nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO` attribute), 374

retry\_count (`nipyapi.nifi.models.versioned_processor.VersionedProcessor` attribute), 525

retry\_count (`nipyapi.registry.models.versioned_processor.VersionedProcessor` attribute), 611

revision (`nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity` attribute), 197

revision (`nipyapi.nifi.models.access_policy_summary_entity.AccessPolicySummaryEntity` attribute), 200

revision (`nipyapi.nifi.models.affected_component_entity.AffectedComponentEntity` attribute), 208

revision (`nipyapi.nifi.models.bucket.Bucket` attribute), 215

revision (`nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity` attribute), 232

revision (`nipyapi.nifi.models.connection_entity.ConnectionEntity` attribute), 243

revision (`nipyapi.nifi.models.controller_configuration_entity.ControllerConfigurationEntity` attribute), 255

revision (`nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity` attribute), 266

revision (`nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity` attribute), 271

revision (`nipyapi.nifi.models.funnel_entity.FunnelEntity` attribute), 307

revision (`nipyapi.nifi.models.label_entity.LabelEntity` attribute), 315

revision (`nipyapi.nifi.models.port_entity.PortEntity` attribute), 341

revision (`nipyapi.nifi.models.process_group_entity.ProcessGroupEntity` attribute), 359

revision (`nipyapi.nifi.models.processor_entity.ProcessorEntity` attribute), 380

revision (`nipyapi.nifi.models.registry_client_entity.RegistryClientEntity` attribute), 413

revision (`nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity` attribute), 423

revision (`nipyapi.nifi.models.remote_process_group_port_entity.RemoteProcessGroupPortEntity` attribute), 427

revision (`nipyapi.nifi.models.reporting_task_entity.ReportingTaskEntity` attribute), 440

revision (`nipyapi.nifi.models.template_entity.TemplateEntity` attribute), 467

revision (`nipyapi.nifi.models.tenant_entity.TenantEntity` attribute), 470

revision (`nipyapi.nifi.models.user_entity.UserEntity` attribute), 476

revision (`nipyapi.nifi.models.user_group_entity.UserGroupEntity` attribute), 479

revision (`nipyapi.nifi.models.versioned_flow.VersionedFlow` attribute), 501

revision (`nipyapi.registry.models.access_policy.AccessPolicy` attribute), 569

revision (`nipyapi.registry.models.access_policy_summary.AccessPolicySummary` attribute), 570

revision (`nipyapi.registry.models.bucket.Bucket` attribute), 573

revision (`nipyapi.registry.models.tenant.Tenant` attribute), 582

revision (`nipyapi.registry.models.user.User` attribute), 583

revision (`nipyapi.registry.models.user_group.UserGroup` attribute), 585

revision (`nipyapi.registry.models.versioned_flow.VersionedFlow` attribute), 594

RevisionDTO (class in `nipyapi.nifi.models.revision_dto`), 443

revision (`nipyapi.registry.models.node_dto.NodeDTO` attribute), 326

revision\_millis



(nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO processor () (in module  
 attribute), 374 nipyapi.canvas), 13  
 run\_duration\_millis ScheduleComponentsEntity (class in  
 (nipyapi.nifi.models.versioned\_processor.VersionedProcessor nipyapi.nifi.models.schedule\_components\_entity),  
 attribute), 525 444  
 run\_duration\_millis scheduled\_state (nipyapi.nifi.models.versioned\_controller\_service.V  
 (nipyapi.registry.models.versioned\_processor.VersionedProcessor attribute), 499  
 attribute), 611 scheduled\_state (nipyapi.nifi.models.versioned\_port.VersionedPort  
 run\_status (nipyapi.nifi.models.port\_status\_dto.PortStatusDTO attribute), 516  
 attribute), 342 scheduled\_state (nipyapi.nifi.models.versioned\_processor.VersionedP  
 run\_status (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO  
 attribute), 345 scheduled\_state (nipyapi.nifi.models.versioned\_remote\_group\_port.V  
 run\_status (nipyapi.nifi.models.processor\_status\_dto.ProcessorStatusDTO), 529  
 attribute), 382 scheduled\_state (nipyapi.registry.models.versioned\_controller\_servic  
 run\_status (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO  
 attribute), 386 scheduled\_state (nipyapi.registry.models.versioned\_port.VersionedPo  
 running (nipyapi.nifi.models.connectable\_dto.ConnectableDTO attribute), 603  
 attribute), 238 scheduled\_state (nipyapi.registry.models.versioned\_processor.Version  
 running\_count (nipyapi.nifi.models.controller\_dto.ControllerDTO attribute), 611  
 attribute), 257 scheduled\_state (nipyapi.registry.models.versioned\_remote\_group\_po  
 running\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO)  
 attribute), 275 scheduling\_period  
 running\_count (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO  
 attribute), 353 attribute), 374  
 running\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity  
 attribute), 359 (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO  
 attribute), 437  
 scheduling\_period  
 sanitize\_for\_serialization () (nipyapi.nifi.models.versioned\_processor.VersionedProcessor  
 (nipyapi.nifi.api\_client.ApiClient method), attribute), 525  
 534 scheduling\_period  
 sanitize\_for\_serialization () (nipyapi.registry.models.versioned\_processor.VersionedProcessor  
 (nipyapi.registry.api\_client.ApiClient method), attribute), 611  
 620 scheduling\_strategy  
 save\_flow\_ver () (in module nipyapi.versioning), 32 (nipyapi.nifi.models.processor\_config\_dto.ProcessorConfigDTO  
 save\_to\_flow\_registry () attribute), 374  
 (nipyapi.nifi.apis.versions\_api.VersionsApi  
 method), 188 scheduling\_strategy  
 save\_to\_flow\_registry\_with\_http\_info () (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO  
 (nipyapi.nifi.apis.versions\_api.VersionsApi attribute), 437  
 method), 188 scheduling\_strategy  
 schedule\_components () (in module (nipyapi.nifi.models.versioned\_processor.VersionedProcessor  
 nipyapi.canvas), 14 attribute), 525  
 schedule\_components () (nipyapi.registry.models.versioned\_processor.VersionedProcessor  
 (nipyapi.nifi.apis.flow\_api.FlowApi method), attribute), 611  
 89 scope (nipyapi.nifi.models.state\_map\_dto.StateMapDTO  
 attribute), 452  
 schedule\_components\_with\_http\_info () search\_cluster () (nipyapi.nifi.apis.flow\_api.FlowApi  
 (nipyapi.nifi.apis.flow\_api.FlowApi method), method), 90  
 89 search\_cluster\_with\_http\_info ()  
 schedule\_controller () (in module (nipyapi.nifi.apis.flow\_api.FlowApi method),  
 nipyapi.canvas), 17 90  
 schedule\_process\_group () (in module search\_flow () (nipyapi.nifi.apis.flow\_api.FlowApi  
 nipyapi.canvas), 11 method), 90

`search_flow_with_http_info()`  
 (`nipyapi.nifi.apis.flow_api.FlowApi` method),  
 90

`search_results_dto`  
 (`nipyapi.nifi.models.search_results_entity.SearchResultsEntity` attribute), 447

`search_tenants()` (`nipyapi.nifi.apis.tenants_api.TenantsApi` attribute), 369  
 method), 179

`search_tenants_with_http_info()`  
 (`nipyapi.nifi.apis.tenants_api.TenantsApi` method), 179

`search_terms` (`nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDTO` attribute), 408

`searchable_fields`  
 (`nipyapi.nifi.models.provenance_options_dto.ProvenanceOptionsDTO` attribute), 405

`SearchResultsDTO` (class in `nipyapi.nifi.models.search_results_dto`), 445

`SearchResultsEntity` (class in `nipyapi.nifi.models.search_results_entity`), 447

`secure` (`nipyapi.nifi.models.peer_dto.PeerDTO` attribute), 335

`select_header_accept()`  
 (`nipyapi.nifi.api_client.ApiClient` method),  
 535

`select_header_accept()`  
 (`nipyapi.registry.api_client.ApiClient` method),  
 621

`select_header_content_type()`  
 (`nipyapi.nifi.api_client.ApiClient` method),  
 535

`select_header_content_type()`  
 (`nipyapi.registry.api_client.ApiClient` method),  
 621

`selected_relationships`  
 (`nipyapi.nifi.models.connection_dto.ConnectionDTO` attribute), 241

`selected_relationships`  
 (`nipyapi.nifi.models.versioned_connection.VersionedConnection` attribute), 496

`selected_relationships`  
 (`nipyapi.registry.models.versioned_connection.VersionedConnection` attribute), 588

`sensitive` (`nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO` attribute), 391

`sensitive` (`nipyapi.nifi.models.versioned_property_descriptor.VersionedPropertyDescriptor` attribute), 527

`sensitive` (`nipyapi.registry.models.versioned_property_descriptor.VersionedPropertyDescriptor` attribute), 613

`sensitive_dynamic_property_names`  
 (`nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO` attribute), 264

`sensitive_dynamic_property_names`  
 (`nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO` attribute), 375

`sensitive_dynamic_property_names`  
 (`nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO` attribute), 437

`sent` (`nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO` attribute), 369

`sent` (`nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO` attribute), 432

`service_login()` (in module `nipyapi.security`), 24

`service_logout()` (in module `nipyapi.security`), 22

`set_default_header()`  
 (`nipyapi.nifi.api_client.ApiClient` method),  
 535

`set_default_header()`  
 (`nipyapi.registry.api_client.ApiClient` method),  
 621

`set_endpoint()` (in module `nipyapi.utils`), 29

`set_remote_process_group_transmission()`  
 (in module `nipyapi.canvas`), 19

`set_service_auth_token()` (in module `nipyapi.security`), 22

`set_service_ssl_context()` (in module `nipyapi.security`), 23

`singleton()` (in module `nipyapi.nifi.configuration`),  
 535

`singleton()` (in module `nipyapi.registry.configuration`), 621

`site_to_site_secure`  
 (`nipyapi.nifi.models.controller_dto.ControllerDTO` attribute), 258

`SiteToSiteApi` (class in `nipyapi.nifi.apis.site_to_site_api`), 170

`size` (`nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO` attribute), 213

`size` (`nipyapi.nifi.models.batch_size.BatchSize` attribute), 214

`size` (`nipyapi.nifi.models.flow_file_dto.FlowFileDTO` attribute), 301

`size` (`nipyapi.nifi.models.flow_file_summary_dto.FlowFileSummaryDTO` attribute), 303

`size` (`nipyapi.registry.models.batch_size.BatchSize` attribute), 624

`snapshot` (`nipyapi.nifi.models.node_counters_snapshot_dto.NodeCountersSnapshotDTO` attribute), 324

`snapshot` (`nipyapi.nifi.models.node_system_diagnostics_snapshot_dto.NodeSystemDiagnosticsSnapshotDTO` attribute), 334

`snapshot_metadata`  
 (`nipyapi.nifi.models.versioned_flow_snapshot.VersionedFlowSnapshot` attribute), 506

`snapshot_metadata`  
 (`nipyapi.registry.models.versioned_flow_snapshot.VersionedFlowSnapshot` attribute), 624

attribute), 596  
 snippet (nipyapi.nifi.models.instantiate\_template\_request\_entity.InstantiateTemplateRequestEntity), 401  
 attribute), 312  
 snippet (nipyapi.nifi.models.snippet\_entity.SnippetEntity), 449  
 attribute), 449  
 snippet (nipyapi.nifi.models.template\_dto.TemplateDTO), 466  
 attribute), 466  
 snippet\_id (nipyapi.nifi.models.copy\_snippet\_request\_entity.CopySnippetRequestEntity), 277  
 attribute), 277  
 snippet\_id (nipyapi.nifi.models.create\_template\_request\_entity.CreateTemplateRequestEntity), 283  
 attribute), 283  
 SnippetDTO (class in nipyapi.nifi.models.snippet\_dto), 447  
 SnippetEntity (class in nipyapi.nifi.models.snippet\_entity), 449  
 SnippetsApi (class in nipyapi.nifi.apis.snippets\_api), 171  
 source (nipyapi.nifi.models.connection\_dto.ConnectionDTO), 241  
 attribute), 241  
 source (nipyapi.nifi.models.versioned\_connection.VersionedConnection), 496  
 attribute), 496  
 source (nipyapi.registry.models.versioned\_connection.VersionedConnection), 588  
 attribute), 588  
 source\_connection\_identifier (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO), 401  
 attribute), 401  
 source\_group\_id (nipyapi.nifi.models.connection\_entity.ConnectionEntity), 244  
 attribute), 244  
 source\_id (nipyapi.nifi.models.action\_dto.ActionDTO), 203  
 attribute), 203  
 source\_id (nipyapi.nifi.models.action\_entity.ActionEntity), 204  
 attribute), 204  
 source\_id (nipyapi.nifi.models.bulletin\_dto.BulletinDTO), 220  
 attribute), 220  
 source\_id (nipyapi.nifi.models.bulletin\_entity.BulletinEntity), 221  
 attribute), 221  
 source\_id (nipyapi.nifi.models.connection\_entity.ConnectionEntity), 244  
 attribute), 244  
 source\_id (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO), 246  
 attribute), 246  
 source\_id (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO), 251  
 attribute), 251  
 source\_id (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO), 403  
 attribute), 403  
 source\_name (nipyapi.nifi.models.action\_dto.ActionDTO), 203  
 attribute), 203  
 source\_name (nipyapi.nifi.models.bulletin\_dto.BulletinDTO), 220  
 attribute), 220  
 source\_name (nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO), 246  
 attribute), 246  
 source\_name (nipyapi.nifi.models.connection\_status\_snapshot\_dto.ConnectionStatusSnapshotDTO), 251  
 attribute), 251  
 source\_running (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO), 321  
 attribute), 321  
 source\_system\_flow\_file\_id (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO), 401  
 attribute), 401  
 source\_type (nipyapi.nifi.models.action\_dto.ActionDTO), 203  
 attribute), 203  
 source\_type (nipyapi.nifi.models.connection\_entity.ConnectionEntity), 244  
 attribute), 244  
 source\_type (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO), 275  
 attribute), 275  
 source\_type (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO), 353  
 attribute), 353  
 stale\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity), 359  
 attribute), 359  
 start\_date (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO), 408  
 attribute), 408  
 start\_docker\_containers () (in module nipyapi.utils), 30  
 start\_version\_control\_request\_entity (class in nipyapi.nifi.models.start\_version\_control\_request\_entity), 450  
 state (nipyapi.nifi.models.activate\_controller\_services\_entity.ActivateControllerServicesEntity), 205  
 attribute), 205  
 state (nipyapi.nifi.models.affected\_component\_dto.AffectedComponentDTO), 207  
 attribute), 207  
 state (nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO), 264  
 attribute), 264  
 state (nipyapi.nifi.models.controller\_service\_referencing\_component\_dto.ControllerServiceReferencingComponentDTO), 268  
 attribute), 268  
 state (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO), 291  
 attribute), 291  
 state (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO), 321  
 attribute), 321  
 state (nipyapi.nifi.models.port\_dto.PortDTO), 339  
 attribute), 339  
 state (nipyapi.nifi.models.processor\_dto.ProcessorDTO), 378  
 attribute), 378  
 state (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO), 438  
 attribute), 438  
 state (nipyapi.nifi.models.schedule\_components\_entity.ScheduleComponentsEntity), 444  
 attribute), 444  
 state (nipyapi.nifi.models.state\_map\_dto.StateMapDTO), 452  
 attribute), 452  
 state (nipyapi.nifi.models.update\_controller\_service\_reference\_request\_dto.UpdateControllerServiceReferenceRequestDTO), 473  
 attribute), 473  
 state (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO), 485  
 attribute), 485  
 state (nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO), 490  
 attribute), 490  
 state (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO), 510  
 attribute), 510  
 state (nipyapi.nifi.models.component\_state\_dto.ComponentStateDTO), 335  
 attribute), 335  
 state\_explanation

(*nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInfo* attribute), 490

(*nipyapi.nifi.models.node\_remote\_process\_group\_status\_dto.NodeRemoteProcessGroupStatusDTO* attribute), 331

*StateEntryDTO* (class in *status\_snapshots* (*nipyapi.nifi.models.node\_status\_snapshots\_dto.NodeStatusSnapshotsDTO* attribute), 333

*StateMapDTO* (class in *StatusDescriptorDTO* (class in *nipyapi.nifi.models.status\_descriptor\_dto*), 452

*stats\_last\_refreshed* (*nipyapi.nifi.models.connection\_status\_dto.ConnectionStatusDTO* attribute), 246

*stats\_last\_refreshed* (*StatusHistoryEntity* (class in *nipyapi.nifi.models.status\_history\_entity*), 455

*stats\_last\_refreshed* (*StatusSnapshotDTO* (class in *nipyapi.nifi.models.status\_snapshot\_dto*), 455

*stats\_last\_refreshed* (*stop\_version\_control()* (*nipyapi.nifi.apis.versions\_api.VersionsApi* method), 188

*stats\_last\_refreshed* (*stop\_version\_control\_with\_http\_info()* (*nipyapi.nifi.apis.versions\_api.VersionsApi* method), 189

*stats\_last\_refreshed* (*stopped\_count* (*nipyapi.nifi.models.controller\_dto.ControllerDTO* attribute), 275

*stats\_last\_refreshed* (*stopped\_count* (*nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO* attribute), 275

*status* (*nipyapi.nifi.models.access\_status\_dto.AccessStatusDTO* attribute), 201

*status* (*nipyapi.nifi.models.connection\_entity.ConnectionEntity* attribute), 244

*status* (*nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity* attribute), 266

*status* (*nipyapi.nifi.models.node\_dto.NodeDTO* attribute), 326

*status* (*nipyapi.nifi.models.port\_entity.PortEntity* attribute), 341

*status* (*nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity* attribute), 359

*status* (*nipyapi.nifi.models.processor\_entity.ProcessorEntity* attribute), 380

*status* (*nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity* attribute), 423

*status* (*nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity* attribute), 440

*status\_history* (*nipyapi.nifi.models.status\_history\_entity.StatusHistoryEntity* attribute), 455

*status\_metrics* (*nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO* attribute), 456

*status\_snapshot* (*nipyapi.nifi.models.node\_connection\_status\_snapshot\_dto.NodeConnectionStatusSnapshotDTO* attribute), 323

*status\_snapshot* (*nipyapi.nifi.models.node\_port\_status\_snapshot\_dto.NodePortStatusSnapshotDTO* attribute), 328

*status\_snapshot* (*nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto.NodeProcessGroupStatusSnapshotDTO* attribute), 329

*status\_snapshot* (*nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.NodeProcessorStatusSnapshotDTO* attribute), 330



`attribute`), 485  
`submit_config_verification_request()` (`nipyapi.nifi.apis.controller_services_api.ControllerServicesApi` (class in `nipyapi.nifi.models.submit_replay_request_entity`), method), 57  
`submit_config_verification_request()` (`nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi` (class in `nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDto`), method), 167  
`submit_config_verification_request_with_http_info()` (`nipyapi.nifi.apis.controller_services_api.ControllerServicesApi` (class in `nipyapi.nifi.models.processor_dto.ProcessorDTO`), method), 57  
`submit_config_verification_request_with_http_info()` (`nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi` (class in `nipyapi.nifi.models.processor_dto.ProcessorDTO`), method), 167  
`submit_lineage_request()` (`nipyapi.nifi.apis.provenance_api.ProvenanceApi` (class in `nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO`), method), 151  
`submit_lineage_request_with_http_info()` (`nipyapi.nifi.apis.provenance_api.ProvenanceApi` (class in `nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO`), method), 152  
`submit_parameter_context_update()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` (class in `nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDto`), method), 110  
`submit_parameter_context_update_with_http_info()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` (class in `nipyapi.nifi.models.processor_dto.ProcessorDTO`), method), 110  
`submit_processor_verification_request()` (`nipyapi.nifi.apis.processors_api.ProcessorsApi` (class in `nipyapi.nifi.models.access_configuration_dto.AccessConfigurationDto`), method), 146  
`submit_processor_verification_request_with_http_info()` (`nipyapi.nifi.apis.processors_api.ProcessorsApi` (class in `nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO`), method), 146  
`submit_provenance_request()` (`nipyapi.nifi.apis.provenance_api.ProvenanceApi` (class in `nipyapi.nifi.models.processor_dto.ProcessorDTO`), method), 152  
`submit_provenance_request_with_http_info()` (`nipyapi.nifi.apis.provenance_api.ProvenanceApi` (class in `nipyapi.nifi.models.reporting_task_dto.ReportingTaskDTO`), method), 152  
`submit_replay()` (`nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi` (class in `nipyapi.nifi.models.about_dto.AboutDTO`), method), 154  
`submit_replay_with_http_info()` (`nipyapi.nifi.apis.provenance_events_api.ProvenanceEventsApi` (class in `nipyapi.nifi.models.about_entity.AboutEntity`), method), 155  
`submit_update_variable_registry_request()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` (class in `nipyapi.nifi.models.access_configuration_dto.AccessConfigurationDto`), method), 136  
`submit_update_variable_registry_request_with_http_info()` (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi` (class in `nipyapi.nifi.models.access_policy_dto.AccessPolicyDTO`), method), 137  
`submit_validation_request()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` (class in `nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity`), method), 111  
`submit_validation_request_with_http_info()` (`nipyapi.nifi.apis.parameter_contexts_api.ParameterContextsApi` (class in `nipyapi.nifi.models.access_policy_summary_dto.AccessPolicySummaryDto`), method), 111



swagger_types (nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO attribute), 264	swagger_types (nipyapi.nifi.models.flow_configuration_entity.FlowConfigurationEntity attribute), 296
swagger_types (nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity attribute), 266	swagger_types (nipyapi.nifi.models.flow_dto.FlowDTO attribute), 298
swagger_types (nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity attribute), 269	swagger_types (nipyapi.nifi.models.flow_dto.FlowDTO attribute), 298
swagger_types (nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity attribute), 271	swagger_types (nipyapi.nifi.models.flow_dto.FlowDTO attribute), 301
swagger_types (nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity attribute), 272	swagger_types (nipyapi.nifi.models.flow_dto.FlowDTO attribute), 301
swagger_types (nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity attribute), 272	swagger_types (nipyapi.nifi.models.flow_dto.FlowDTO attribute), 303
swagger_types (nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity attribute), 273	swagger_types (nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO attribute), 305
swagger_types (nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO attribute), 275	swagger_types (nipyapi.nifi.models.funnel_dto.FunnelDTO attribute), 306
swagger_types (nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity attribute), 276	swagger_types (nipyapi.nifi.models.funnel_entity.FunnelEntity attribute), 307
swagger_types (nipyapi.nifi.models.copy_snippet_request_entity.CopySnippetRequestEntity attribute), 277	swagger_types (nipyapi.nifi.models.funnels_entity.FunnelsEntity attribute), 308
swagger_types (nipyapi.nifi.models.counter_dto.CounterDTO attribute), 278	swagger_types (nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO attribute), 309
swagger_types (nipyapi.nifi.models.counter_entity.CounterEntity attribute), 279	swagger_types (nipyapi.nifi.models.history_dto.HistoryDTO attribute), 310
swagger_types (nipyapi.nifi.models.counters_dto.CountersDTO attribute), 280	swagger_types (nipyapi.nifi.models.history_entity.HistoryEntity attribute), 310
swagger_types (nipyapi.nifi.models.counters_entity.CountersEntity attribute), 280	swagger_types (nipyapi.nifi.models.input_ports_entity.InputPortsEntity attribute), 311
swagger_types (nipyapi.nifi.models.counters_snapshot_dto.CountersSnapshotDTO attribute), 281	swagger_types (nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity attribute), 312
swagger_types (nipyapi.nifi.models.create_active_request_entity.CreateActiveRequestEntity attribute), 282	swagger_types (nipyapi.nifi.models.label_dto.LabelDTO attribute), 313
swagger_types (nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity attribute), 283	swagger_types (nipyapi.nifi.models.label_entity.LabelEntity attribute), 315
swagger_types (nipyapi.nifi.models.current_user_entity.CurrentUserEntity attribute), 285	swagger_types (nipyapi.nifi.models.labels_entity.LabelsEntity attribute), 316
swagger_types (nipyapi.nifi.models.difference_dto.DifferenceDTO attribute), 286	swagger_types (nipyapi.nifi.models.lineage_dto.LineageDTO attribute), 317
swagger_types (nipyapi.nifi.models.dimensions_dto.DimensionsDTO attribute), 287	swagger_types (nipyapi.nifi.models.lineage_entity.LineageEntity attribute), 318
swagger_types (nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO attribute), 288	swagger_types (nipyapi.nifi.models.lineage_request_dto.LineageRequestDTO attribute), 319
swagger_types (nipyapi.nifi.models.drop_request_dto.DropRequestDTO attribute), 291	swagger_types (nipyapi.nifi.models.lineage_results_dto.LineageResultsDTO attribute), 320
swagger_types (nipyapi.nifi.models.drop_request_entity.DropRequestEntity attribute), 291	swagger_types (nipyapi.nifi.models.listing_request_dto.ListingRequestDTO attribute), 322
swagger_types (nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO attribute), 292	swagger_types (nipyapi.nifi.models.listing_request_entity.ListingRequestEntity attribute), 322
swagger_types (nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity attribute), 293	swagger_types (nipyapi.nifi.models.node_connection_status_snapshot_dto.NodeConnectionStatusSnapshotDTO attribute), 323
swagger_types (nipyapi.nifi.models.flow_comparison_entity.FlowComparisonEntity attribute), 294	swagger_types (nipyapi.nifi.models.node_counters_snapshot_dto.NodeCountersSnapshotDTO attribute), 324
swagger_types (nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO attribute), 296	swagger_types (nipyapi.nifi.models.node_dto.NodeDTO attribute), 326

swagger\_types (nipyapi.nifi.models.node\_entity.NodeEntity attribute), 327

swagger\_types (nipyapi.nifi.models.node\_event\_dto.NodeEventDTO attribute), 327

swagger\_types (nipyapi.nifi.models.node\_port\_status\_snapshot\_dto.NodePortStatusSnapshotDTO attribute), 328

swagger\_types (nipyapi.nifi.models.node\_process\_group\_status\_snapshot\_dto.NodeProcessGroupStatusSnapshotDTO attribute), 329

swagger\_types (nipyapi.nifi.models.node\_processor\_status\_snapshot\_dto.NodeProcessorStatusSnapshotDTO attribute), 330

swagger\_types (nipyapi.nifi.models.node\_remote\_process\_group\_status\_snapshot\_dto.NodeRemoteProcessGroupStatusSnapshotDTO attribute), 331

swagger\_types (nipyapi.nifi.models.node\_search\_results\_dto.NodeSearchResultsDTO attribute), 332

swagger\_types (nipyapi.nifi.models.node\_status\_snapshot\_dto.NodeStatusSnapshotDTO attribute), 333

swagger\_types (nipyapi.nifi.models.node\_system\_diagnostics\_snapshot\_dto.NodeSystemDiagnosticsSnapshotDTO attribute), 334

swagger\_types (nipyapi.nifi.models.output\_ports\_entity.OutputPortsEntity attribute), 335

swagger\_types (nipyapi.nifi.models.peer\_dto.PeerDTO attribute), 335

swagger\_types (nipyapi.nifi.models.peers\_entity.PeersEntity attribute), 336

swagger\_types (nipyapi.nifi.models.permissions.Permissions attribute), 337

swagger\_types (nipyapi.nifi.models.permissions\_dto.PermissionsDTO attribute), 337

swagger\_types (nipyapi.nifi.models.port\_dto.PortDTO attribute), 339

swagger\_types (nipyapi.nifi.models.port\_entity.PortEntity attribute), 341

swagger\_types (nipyapi.nifi.models.port\_status\_dto.PortStatusDTO attribute), 343

swagger\_types (nipyapi.nifi.models.port\_status\_entity.PortStatusEntity attribute), 344

swagger\_types (nipyapi.nifi.models.port\_status\_snapshot\_dto.PortStatusSnapshotDTO attribute), 345

swagger\_types (nipyapi.nifi.models.port\_status\_snapshot\_entity.PortStatusSnapshotEntity attribute), 346

swagger\_types (nipyapi.nifi.models.position\_dto.PositionDTO attribute), 347

swagger\_types (nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO attribute), 348

swagger\_types (nipyapi.nifi.models.prioritizer\_types\_entity.PrioritizerTypesEntity attribute), 348

swagger\_types (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO attribute), 354

swagger\_types (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 359

swagger\_types (nipyapi.nifi.models.process\_group\_flows\_dto.ProcessGroupFlowsDTO attribute), 361

swagger\_types (nipyapi.nifi.models.process\_group\_flows\_entity.ProcessGroupFlowsEntity attribute), 362

swagger\_types (nipyapi.nifi.models.process\_group\_status\_dto.ProcessGroupStatusDTO attribute), 363

swagger\_types (nipyapi.nifi.models.process\_group\_status\_entity.ProcessGroupStatusEntity attribute), 364

swagger\_types (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 369

swagger\_types (nipyapi.nifi.models.process\_group\_status\_snapshot\_entity.ProcessGroupStatusSnapshotEntity attribute), 371

swagger\_types (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 371

swagger\_types (nipyapi.nifi.models.processor\_entity.ProcessorEntity attribute), 371

swagger\_types (nipyapi.nifi.models.processor\_report\_config\_status\_snapshot\_dto.ProcessorReportConfigStatusSnapshotDTO attribute), 375

swagger\_types (nipyapi.nifi.models.processor\_dto.ProcessorDTO attribute), 378

swagger\_types (nipyapi.nifi.models.processor\_entity.ProcessorEntity attribute), 381

swagger\_types (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 382

swagger\_types (nipyapi.nifi.models.processor\_status\_entity.ProcessorStatusEntity attribute), 383

swagger\_types (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 386

swagger\_types (nipyapi.nifi.models.processor\_status\_snapshot\_entity.ProcessorStatusSnapshotEntity attribute), 388

swagger\_types (nipyapi.nifi.models.processor\_types\_entity.ProcessorTypesEntity attribute), 388

swagger\_types (nipyapi.nifi.models.processors\_entity.ProcessorsEntity attribute), 389

swagger\_types (nipyapi.nifi.models.property\_descriptor\_dto.PropertyDescriptorDTO attribute), 392

swagger\_types (nipyapi.nifi.models.property\_descriptor\_entity.PropertyDescriptorEntity attribute), 392

swagger\_types (nipyapi.nifi.models.property\_history\_dto.PropertyHistoryDTO attribute), 393

swagger\_types (nipyapi.nifi.models.provenance\_dto.ProvenanceDTO attribute), 394

swagger\_types (nipyapi.nifi.models.provenance\_entity.ProvenanceEntity attribute), 395

swagger\_types (nipyapi.nifi.models.provenance\_event\_dto.ProvenanceEventDTO attribute), 401

swagger\_types (nipyapi.nifi.models.provenance\_event\_entity.ProvenanceEventEntity attribute), 402

swagger\_types (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO attribute), 403

swagger\_types (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO attribute), 405

swagger\_types (nipyapi.nifi.models.provenance\_options\_dto.ProvenanceOptionsDTO attribute), 405

swagger\_types (nipyapi.nifi.models.provenance\_options\_entity.ProvenanceOptionsEntity attribute), 406

swagger\_types (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO attribute), 408

swagger\_types (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO attribute), 409



swagger\_types (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO\_entity.SnippetEntity attribute), 410

swagger\_types (nipyapi.nifi.models.queue\_size\_dto.QueueSizeDTO\_entity.SnippetEntity attribute), 411

swagger\_types (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity attribute), 413

swagger\_types (nipyapi.nifi.models.registry\_clients\_entity.RegistryClientsEntity attribute), 413

swagger\_types (nipyapi.nifi.models.registry\_dto.RegistryDTO\_entity.SnippetEntity attribute), 414

swagger\_types (nipyapi.nifi.models.relationship\_dto.RelationshipDTO\_entity.SnippetEntity attribute), 415

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_content\_dto.RemoteProcessGroupContentDTO\_entity.StatusHistory attribute), 416

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO\_entity.SnippetEntity attribute), 420

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity attribute), 423

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_port\_dto.RemoteProcessGroupPortDTO\_entity.SnippetEntity attribute), 425

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity attribute), 427

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO\_entity.SnippetEntity attribute), 429

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity attribute), 430

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_status\_swaps\_entity.RemoteProcessGroupStatusSwapsEntity attribute), 432

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_status\_swaps\_entity.RemoteProcessGroupStatusSwapsEntity attribute), 433

swagger\_types (nipyapi.nifi.models.remote\_process\_group\_status\_swaps\_entity.RemoteProcessGroupStatusSwapsEntity attribute), 434

swagger\_types (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO\_entity.SnippetEntity attribute), 438

swagger\_types (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity attribute), 440

swagger\_types (nipyapi.nifi.models.reporting\_task\_types\_entity.ReportingTaskTypesEntity attribute), 441

swagger\_types (nipyapi.nifi.models.reporting\_tasks\_entity.ReportingTasksEntity attribute), 441

swagger\_types (nipyapi.nifi.models.resource\_dto.ResourceDTO\_entity.SnippetEntity attribute), 442

swagger\_types (nipyapi.nifi.models.resources\_entity.ResourcesEntity attribute), 442

swagger\_types (nipyapi.nifi.models.revision\_dto.RevisionDTO\_entity.SnippetEntity attribute), 443

swagger\_types (nipyapi.nifi.models.schedule\_components\_entity.ScheduleComponentsEntity attribute), 444

swagger\_types (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO\_entity.SnippetEntity attribute), 446

swagger\_types (nipyapi.nifi.models.search\_results\_entity.SearchResultsEntity attribute), 447

swagger\_types (nipyapi.nifi.models.snippet\_dto.SnippetDTO\_entity.SnippetEntity attribute), 449

swagger\_types (nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity attribute), 450

swagger\_types (nipyapi.nifi.models.state\_entry\_dto.StateEntryDTO\_entity.SnippetEntity attribute), 451

swagger\_types (nipyapi.nifi.models.state\_map\_dto.StateMapDTO\_entity.SnippetEntity attribute), 452

swagger\_types (nipyapi.nifi.models.status\_descriptor\_dto.StatusDescriptorDTO\_entity.SnippetEntity attribute), 453

swagger\_types (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO\_entity.SnippetEntity attribute), 454

swagger\_types (nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO\_entity.SnippetEntity attribute), 455

swagger\_types (nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO\_entity.SnippetEntity attribute), 456

swagger\_types (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO\_entity.SnippetEntity attribute), 457

swagger\_types (nipyapi.nifi.models.streaming\_output\_entity.StreamingOutputEntity attribute), 458

swagger\_types (nipyapi.nifi.models.submission\_request\_entity.SubmissionRequestEntity attribute), 458

swagger\_types (nipyapi.nifi.models.system\_diagnostics\_dto.SystemDiagnosticsDTO\_entity.SnippetEntity attribute), 459

swagger\_types (nipyapi.nifi.models.system\_diagnostics\_entity.SystemDiagnosticsEntity attribute), 460

swagger\_types (nipyapi.nifi.models.system\_diagnostics\_entity.SystemDiagnosticsEntity attribute), 463

swagger\_types (nipyapi.nifi.models.system\_diagnostics\_entity.SystemDiagnosticsEntity attribute), 466

swagger\_types (nipyapi.nifi.models.template\_entity.TemplateEntity attribute), 467

swagger\_types (nipyapi.nifi.models.templates\_entity.TemplatesEntity attribute), 468

swagger\_types (nipyapi.nifi.models.tenant\_dto.TenantDTO\_entity.SnippetEntity attribute), 469

swagger\_types (nipyapi.nifi.models.tenant\_entity.TenantEntity attribute), 471

swagger\_types (nipyapi.nifi.models.tenants\_entity.TenantsEntity attribute), 471

swagger\_types (nipyapi.nifi.models.transaction\_result\_entity.TransactionResultEntity attribute), 472

swagger\_types (nipyapi.nifi.models.update\_controller\_service\_reference\_entity.UpdateControllerServiceReferenceEntity attribute), 473

swagger\_types (nipyapi.nifi.models.user\_dto.UserDTO\_entity.SnippetEntity attribute), 475

swagger\_types (nipyapi.nifi.models.user\_entity.UserEntity attribute), 476

swagger\_types (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO\_entity.SnippetEntity attribute), 477

swagger\_types (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity attribute), 479

swagger\_types (nipyapi.nifi.models.user\_groups\_entity.UserGroupsEntity attribute), 480



T

tenants\_permissions (nipyapi.nifi.models.current\_user\_entity.CurrentUserEntity attribute), 598  
 attribute), 285  
 TenantsApi (class in nipyapi.nifi.apis.tenants\_api), 175  
 TenantsApi (class in nipyapi.registry.apis.tenants\_api), 562  
 TenantsEntity (class in nipyapi.nifi.models.tenants\_entity), 471  
 terminate\_processor() (nipyapi.nifi.apis.processors\_api.ProcessorsApi method), 147  
 terminate\_processor\_with\_http\_info() (nipyapi.nifi.apis.processors\_api.ProcessorsApi method), 147  
 terminated\_thread\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO attribute), 276  
 terminated\_thread\_count (nipyapi.nifi.models.process\_group\_status\_snapshot\_dto.ProcessGroupStatusSnapshotDTO attribute), 369  
 terminated\_thread\_count (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO attribute), 386  
 test\_identity\_provider\_recognizes\_credentials\_for() (nipyapi.registry.apis.access\_api.AccessApi method), 542  
 test\_identity\_provider\_recognizes\_credentials\_from() (nipyapi.registry.apis.access\_api.AccessApi method), 543  
 time\_offset (nipyapi.nifi.models.flow\_configuration\_dto.FlowConfigurationDTO attribute), 296  
 time\_offset (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO attribute), 409  
 timestamp (nipyapi.nifi.models.action\_dto.ActionDTO attribute), 203  
 timestamp (nipyapi.nifi.models.action\_entity.ActionEntity attribute), 204  
 timestamp (nipyapi.nifi.models.bulletin\_dto.BulletinDTO attribute), 220  
 timestamp (nipyapi.nifi.models.bulletin\_entity.BulletinEntity attribute), 222  
 timestamp (nipyapi.nifi.models.node\_event\_dto.NodeEventDTO attribute), 327  
 timestamp (nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO attribute), 348  
 timestamp (nipyapi.nifi.models.provenance\_link\_dto.ProvenanceLinkDTO attribute), 403  
 timestamp (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO attribute), 405  
 timestamp (nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO attribute), 456  
 timestamp (nipyapi.nifi.models.template\_dto.TemplateDTO attribute), 466  
 timestamp (nipyapi.registry.models.versioned\_flow\_snapshot\_metadata attribute), 598  
 timezone (nipyapi.nifi.models.about\_dto>AboutDTO attribute), 191  
 title (nipyapi.nifi.models.about\_dto>AboutDTO attribute), 191  
 to\_dict() (nipyapi.nifi.models.about\_dto>AboutDTO method), 191  
 to\_dict() (nipyapi.nifi.models.about\_entity>AboutEntity method), 192  
 to\_dict() (nipyapi.nifi.models.access\_configuration\_dto.AccessConfigurationDTO method), 193  
 to\_dict() (nipyapi.nifi.models.access\_configuration\_entity.AccessConfigurationEntity method), 193  
 to\_dict() (nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO method), 195  
 to\_dict() (nipyapi.nifi.models.access\_policy\_entity.AccessPolicyEntity method), 197  
 to\_dict() (nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO method), 198  
 to\_dict() (nipyapi.nifi.models.access\_policy\_summary\_entity.AccessPolicySummaryEntity method), 200  
 to\_dict() (nipyapi.nifi.models.access\_status\_dto.AccessStatusDTO method), 201  
 to\_dict() (nipyapi.nifi.models.access\_status\_entity.AccessStatusEntity method), 201  
 to\_dict() (nipyapi.nifi.models.action\_details\_dto.ActionDetailsDTO method), 202  
 to\_dict() (nipyapi.nifi.models.action\_dto>ActionDTO method), 203  
 to\_dict() (nipyapi.nifi.models.action\_entity>ActionEntity method), 204  
 to\_dict() (nipyapi.nifi.models.activate\_controller\_services\_entity.ActivateControllerServicesEntity method), 205  
 to\_dict() (nipyapi.nifi.models.affected\_component\_dto>AffectedComponentDTO method), 207  
 to\_dict() (nipyapi.nifi.models.affected\_component\_entity>AffectedComponentEntity method), 209  
 to\_dict() (nipyapi.nifi.models.allowable\_value\_dto>AllowableValueDTO method), 209  
 to\_dict() (nipyapi.nifi.models.allowable\_value\_entity>AllowableValueEntity method), 210  
 to\_dict() (nipyapi.nifi.models.attribute\_dto>AttributeDTO method), 211  
 to\_dict() (nipyapi.nifi.models.banner\_dto>BannerDTO method), 212  
 to\_dict() (nipyapi.nifi.models.banner\_entity>BannerEntity method), 212  
 to\_dict() (nipyapi.nifi.models.batch\_settings\_dto>BatchSettingsDTO method), 213  
 to\_dict() (nipyapi.nifi.models.batch\_size>BatchSize method), 214  
 to\_dict() (nipyapi.nifi.models.bucket>Bucket method), 215



`to_dict()` (`nipyapi.nifi.models.bucket_dto.BucketDTO` `to_dict()` (`nipyapi.nifi.models.connection_status_dto.ConnectionStatusDTO`  
`method`), 216 `method`), 246  
`to_dict()` (`nipyapi.nifi.models.bucket_entity.BucketEntity` `to_dict()` (`nipyapi.nifi.models.connection_status_entity.ConnectionStatusEntity`  
`method`), 217 `method`), 247  
`to_dict()` (`nipyapi.nifi.models.buckets_entity.BucketsEntity` `to_dict()` (`nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO`  
`method`), 218 `method`), 251  
`to_dict()` (`nipyapi.nifi.models.bulletin_board_dto.BulletinBoardDTO` `to_dict()` (`nipyapi.nifi.models.connection_status_snapshot_entity.ConnectionStatusSnapshotEntity`  
`method`), 218 `method`), 252  
`to_dict()` (`nipyapi.nifi.models.bulletin_board_entity.BulletinBoardEntity` `to_dict()` (`nipyapi.nifi.models.connections_entity.ConnectionsEntity`  
`method`), 219 `method`), 252  
`to_dict()` (`nipyapi.nifi.models.bulletin_dto.BulletinDTO` `to_dict()` (`nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity`  
`method`), 220 `method`), 253  
`to_dict()` (`nipyapi.nifi.models.bulletin_entity.BulletinEntity` `to_dict()` (`nipyapi.nifi.models.controller_configuration_dto.ControllerConfigurationDTO`  
`method`), 222 `method`), 254  
`to_dict()` (`nipyapi.nifi.models.bundle.Bundle` `to_dict()` (`nipyapi.nifi.models.controller_configuration_entity.ControllerConfigurationEntity`  
`method`), 222 `method`), 255  
`to_dict()` (`nipyapi.nifi.models.bundle_dto.BundleDTO` `to_dict()` (`nipyapi.nifi.models.controller_dto.ControllerDTO`  
`method`), 223 `method`), 258  
`to_dict()` (`nipyapi.nifi.models.cluste_summary_entity.ClusterSummaryEntity` `to_dict()` (`nipyapi.nifi.models.controller_entity.ControllerEntity`  
`method`), 224 `method`), 258  
`to_dict()` (`nipyapi.nifi.models.cluster_dto.ClusterDTO` `to_dict()` (`nipyapi.nifi.models.controller_service_api.ControllerServiceAPI`  
`method`), 224 `method`), 259  
`to_dict()` (`nipyapi.nifi.models.cluster_entity.ClusterEntity` `to_dict()` (`nipyapi.nifi.models.controller_service_api_dto.ControllerServiceAPIDTO`  
`method`), 225 `method`), 260  
`to_dict()` (`nipyapi.nifi.models.cluster_search_results_entity.ClusterSearchResultsEntity` `to_dict()` (`nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO`  
`method`), 225 `method`), 264  
`to_dict()` (`nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO` `to_dict()` (`nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity`  
`method`), 227 `method`), 266  
`to_dict()` (`nipyapi.nifi.models.component_details_dto.ComponentDetailsDTO` `to_dict()` (`nipyapi.nifi.models.controller_service_referencing_component_dto.ControllerServiceReferencingComponentDTO`  
`method`), 227 `method`), 269  
`to_dict()` (`nipyapi.nifi.models.component_difference_dto.ComponentDifferenceDTO` `to_dict()` (`nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity`  
`method`), 228 `method`), 271  
`to_dict()` (`nipyapi.nifi.models.component_history_dto.ComponentHistoryDTO` `to_dict()` (`nipyapi.nifi.models.controller_service_referencing_component_entity_dto.ControllerServiceReferencingComponentEntityDTO`  
`method`), 229 `method`), 272  
`to_dict()` (`nipyapi.nifi.models.component_history_entity.ComponentHistoryEntity` `to_dict()` (`nipyapi.nifi.models.controller_service_types_entity.ControllerServiceTypesEntity`  
`method`), 230 `method`), 272  
`to_dict()` (`nipyapi.nifi.models.component_reference_dto.ComponentReferenceDTO` `to_dict()` (`nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity`  
`method`), 231 `method`), 273  
`to_dict()` (`nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity` `to_dict()` (`nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO`  
`method`), 232 `method`), 276  
`to_dict()` (`nipyapi.nifi.models.component_search_result_dto.ComponentSearchResultDTO` `to_dict()` (`nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity`  
`method`), 234 `method`), 276  
`to_dict()` (`nipyapi.nifi.models.component_state_dto.ComponentStateDTO` `to_dict()` (`nipyapi.nifi.models.copy_snippet_request_entity.CopySnippetRequestEntity`  
`method`), 235 `method`), 277  
`to_dict()` (`nipyapi.nifi.models.component_state_entity.ComponentStateEntity` `to_dict()` (`nipyapi.nifi.models.counter_dto.CounterDTO`  
`method`), 235 `method`), 278  
`to_dict()` (`nipyapi.nifi.models.connectable_component.ConnectableComponent` `to_dict()` (`nipyapi.nifi.models.counter_entity.CounterEntity`  
`method`), 236 `method`), 279  
`to_dict()` (`nipyapi.nifi.models.connectable_dto.ConnectableDTO` `to_dict()` (`nipyapi.nifi.models.counters_dto.CountersDTO`  
`method`), 238 `method`), 280  
`to_dict()` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` `to_dict()` (`nipyapi.nifi.models.counters_entity.CountersEntity`  
`method`), 241 `method`), 280  
`to_dict()` (`nipyapi.nifi.models.connection_entity.ConnectionEntity` `to_dict()` (`nipyapi.nifi.models.counters_snapshot_dto.CountersSnapshotDTO`  
`method`), 244 `method`), 281

`to_dict()` (`nipyapi.nifi.models.create_active_request_entity.CreateActiveRequestEntity` method), 282  
`to_dict()` (`nipyapi.nifi.models.create_active_request_entity.CreateActiveRequestEntity` method), 313  
`to_dict()` (`nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity` method), 283  
`to_dict()` (`nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity` method), 315  
`to_dict()` (`nipyapi.nifi.models.current_user_entity.CurrentUserEntity` method), 285  
`to_dict()` (`nipyapi.nifi.models.current_user_entity.CurrentUserEntity` method), 316  
`to_dict()` (`nipyapi.nifi.models.difference_dto.DifferenceDTO` method), 286  
`to_dict()` (`nipyapi.nifi.models.difference_dto.DifferenceDTO` method), 317  
`to_dict()` (`nipyapi.nifi.models.dimensions_dto.DimensionsDTO` method), 287  
`to_dict()` (`nipyapi.nifi.models.dimensions_dto.DimensionsDTO` method), 318  
`to_dict()` (`nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO` method), 288  
`to_dict()` (`nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO` method), 319  
`to_dict()` (`nipyapi.nifi.models.drop_request_dto.DropRequestDTO` method), 291  
`to_dict()` (`nipyapi.nifi.models.drop_request_dto.DropRequestDTO` method), 320  
`to_dict()` (`nipyapi.nifi.models.drop_request_entity.DropRequestEntity` method), 291  
`to_dict()` (`nipyapi.nifi.models.drop_request_entity.DropRequestEntity` method), 322  
`to_dict()` (`nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO` method), 292  
`to_dict()` (`nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO` method), 322  
`to_dict()` (`nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity` method), 293  
`to_dict()` (`nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity` method), 323  
`to_dict()` (`nipyapi.nifi.models.flow_comparison_entity.FlowComparisonEntity` method), 294  
`to_dict()` (`nipyapi.nifi.models.flow_comparison_entity.FlowComparisonEntity` method), 324  
`to_dict()` (`nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO` method), 296  
`to_dict()` (`nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO` method), 326  
`to_dict()` (`nipyapi.nifi.models.flow_configuration_entity.FlowConfigurationEntity` method), 296  
`to_dict()` (`nipyapi.nifi.models.flow_configuration_entity.FlowConfigurationEntity` method), 327  
`to_dict()` (`nipyapi.nifi.models.flow_dto.FlowDTO` method), 298  
`to_dict()` (`nipyapi.nifi.models.flow_dto.FlowDTO` method), 327  
`to_dict()` (`nipyapi.nifi.models.flow_entity.FlowEntity` method), 298  
`to_dict()` (`nipyapi.nifi.models.flow_entity.FlowEntity` method), 328  
`to_dict()` (`nipyapi.nifi.models.flow_file_dto.FlowFileDTO` method), 301  
`to_dict()` (`nipyapi.nifi.models.flow_file_dto.FlowFileDTO` method), 329  
`to_dict()` (`nipyapi.nifi.models.flow_file_entity.FlowFileEntity` method), 301  
`to_dict()` (`nipyapi.nifi.models.flow_file_entity.FlowFileEntity` method), 330  
`to_dict()` (`nipyapi.nifi.models.flow_file_summary_dto.FlowFileSummaryDTO` method), 303  
`to_dict()` (`nipyapi.nifi.models.flow_file_summary_dto.FlowFileSummaryDTO` method), 331  
`to_dict()` (`nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO` method), 305  
`to_dict()` (`nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO` method), 332  
`to_dict()` (`nipyapi.nifi.models.funnel_dto.FunnelDTO` method), 306  
`to_dict()` (`nipyapi.nifi.models.funnel_dto.FunnelDTO` method), 333  
`to_dict()` (`nipyapi.nifi.models.funnel_entity.FunnelEntity` method), 307  
`to_dict()` (`nipyapi.nifi.models.funnel_entity.FunnelEntity` method), 334  
`to_dict()` (`nipyapi.nifi.models.funnels_entity.FunnelsEntity` method), 308  
`to_dict()` (`nipyapi.nifi.models.funnels_entity.FunnelsEntity` method), 335  
`to_dict()` (`nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO` method), 309  
`to_dict()` (`nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO` method), 335  
`to_dict()` (`nipyapi.nifi.models.history_dto.HistoryDTO` method), 310  
`to_dict()` (`nipyapi.nifi.models.history_dto.HistoryDTO` method), 336  
`to_dict()` (`nipyapi.nifi.models.history_entity.HistoryEntity` method), 310  
`to_dict()` (`nipyapi.nifi.models.history_entity.HistoryEntity` method), 337  
`to_dict()` (`nipyapi.nifi.models.input_ports_entity.InputPortsEntity` method), 311  
`to_dict()` (`nipyapi.nifi.models.input_ports_entity.InputPortsEntity` method), 337  
`to_dict()` (`nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity` method), 312  
`to_dict()` (`nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity` method), 339

---

```

to_dict() (nipyapi.nifi.models.port_entity.PortEntity
method), 341
to_dict() (nipyapi.nifi.models.port_status_dto.PortStatusDTO
method), 343
to_dict() (nipyapi.nifi.models.port_status_entity.PortStatusEntity
method), 344
to_dict() (nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO
method), 346
to_dict() (nipyapi.nifi.models.port_status_snapshot_entity.PortStatusSnapshotEntity
method), 346
to_dict() (nipyapi.nifi.models.position_dto.PositionDTO
method), 347
to_dict() (nipyapi.nifi.models.previous_value_dto.PreviousValueDTO
method), 348
to_dict() (nipyapi.nifi.models.prioritizer_types_entity.PrioritizerTypesEntity
method), 348
to_dict() (nipyapi.nifi.models.process_group_dto.ProcessGroupDTO
method), 354
to_dict() (nipyapi.nifi.models.process_group_entity.ProcessGroupEntity
method), 359
to_dict() (nipyapi.nifi.models.process_group_flow_dto.ProcessGroupFlowDTO
method), 361
to_dict() (nipyapi.nifi.models.process_group_flow_entity.ProcessGroupFlowEntity
method), 362
to_dict() (nipyapi.nifi.models.process_group_status_dto.ProcessGroupStatusDTO
method), 363
to_dict() (nipyapi.nifi.models.process_group_status_entity.ProcessGroupStatusEntity
method), 364
to_dict() (nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO
method), 369
to_dict() (nipyapi.nifi.models.process_group_status_snapshot_entity.ProcessGroupStatusSnapshotEntity
method), 371
to_dict() (nipyapi.nifi.models.process_groups_entity.ProcessGroupsEntity
method), 371
to_dict() (nipyapi.nifi.models.processor_config_dto.ProcessorConfigDTO
method), 375
to_dict() (nipyapi.nifi.models.processor_dto.ProcessorDTO
method), 378
to_dict() (nipyapi.nifi.models.processor_entity.ProcessorEntity
method), 381
to_dict() (nipyapi.nifi.models.processor_status_dto.ProcessorStatusDTO
method), 382
to_dict() (nipyapi.nifi.models.processor_status_entity.ProcessorStatusEntity
method), 383
to_dict() (nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO
method), 387
to_dict() (nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity
method), 388
to_dict() (nipyapi.nifi.models.processor_types_entity.ProcessorTypesEntity
method), 388
to_dict() (nipyapi.nifi.models.processors_entity.ProcessorsEntity
method), 389
to_dict() (nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO
method), 392
to_dict() (nipyapi.nifi.models.property_descriptor_entity.PropertyDescriptorEntity
method), 392
to_dict() (nipyapi.nifi.models.property_history_dto.PropertyHistoryDTO
method), 393
to_dict() (nipyapi.nifi.models.provenance_dto.ProvenanceDTO
method), 394
to_dict() (nipyapi.nifi.models.provenance_entity.ProvenanceEntity
method), 395
to_dict() (nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO
method), 401
to_dict() (nipyapi.nifi.models.provenance_event_entity.ProvenanceEventEntity
method), 402
to_dict() (nipyapi.nifi.models.provenance_link_dto.ProvenanceLinkDTO
method), 403
to_dict() (nipyapi.nifi.models.provenance_node_dto.ProvenanceNodeDTO
method), 405
to_dict() (nipyapi.nifi.models.provenance_options_dto.ProvenanceOptionsDTO
method), 405
to_dict() (nipyapi.nifi.models.provenance_options_entity.ProvenanceOptionsEntity
method), 406
to_dict() (nipyapi.nifi.models.provenance_request_dto.ProvenanceRequestDTO
method), 408
to_dict() (nipyapi.nifi.models.provenance_results_dto.ProvenanceResultsDTO
method), 409
to_dict() (nipyapi.nifi.models.provenance_searchable_field_dto.ProvenanceSearchableFieldDTO
method), 410
to_dict() (nipyapi.nifi.models.queue_size_dto.QueueSizeDTO
method), 411
to_dict() (nipyapi.nifi.models.registry_client_entity.RegistryClientEntity
method), 413
to_dict() (nipyapi.nifi.models.registry_client_entities_entity.RegistryClientEntitiesEntity
method), 413
to_dict() (nipyapi.nifi.models.registry_dto.RegistryDTO
method), 414
to_dict() (nipyapi.nifi.models.relationship_dto.RelationshipDTO
method), 415
to_dict() (nipyapi.nifi.models.remote_process_group_contents_dto.RemoteProcessGroupContentsDTO
method), 416
to_dict() (nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO
method), 420
to_dict() (nipyapi.nifi.models.remote_process_group_entity.RemoteProcessGroupEntity
method), 423
to_dict() (nipyapi.nifi.models.remote_process_group_port_dto.RemoteProcessGroupPortDTO
method), 425
to_dict() (nipyapi.nifi.models.remote_process_group_port_entity.RemoteProcessGroupPortEntity
method), 427
to_dict() (nipyapi.nifi.models.remote_process_group_status_dto.RemoteProcessGroupStatusDTO
method), 429
to_dict() (nipyapi.nifi.models.remote_process_group_status_entity.RemoteProcessGroupStatusEntity
method), 430
to_dict() (nipyapi.nifi.models.remote_process_group_status_snapshot_dto.RemoteProcessGroupStatusSnapshotDTO
method), 432
to_dict() (nipyapi.nifi.models.remote_process_group_status_snapshot_entity.RemoteProcessGroupStatusSnapshotEntity
method), 433

```

`to_dict()` (nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity method), 434  
`to_dict()` (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO method), 438  
`to_dict()` (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity method), 440  
`to_dict()` (nipyapi.nifi.models.reporting\_task\_types\_entity.ReportingTaskTypesEntity method), 441  
`to_dict()` (nipyapi.nifi.models.reporting\_tasks\_entity.ReportingTasksEntity method), 441  
`to_dict()` (nipyapi.nifi.models.resource\_dto.ResourceDTO method), 442  
`to_dict()` (nipyapi.nifi.models.resources\_entity.ResourcesEntity method), 442  
`to_dict()` (nipyapi.nifi.models.revision\_dto.RevisionDTO method), 443  
`to_dict()` (nipyapi.nifi.models.schedule\_components\_entity.ScheduleComponentsEntity method), 444  
`to_dict()` (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO method), 446  
`to_dict()` (nipyapi.nifi.models.search\_results\_entity.SearchResultsEntity method), 447  
`to_dict()` (nipyapi.nifi.models.snippet\_dto.SnippetDTO method), 449  
`to_dict()` (nipyapi.nifi.models.snippet\_entity.SnippetEntity method), 450  
`to_dict()` (nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity method), 450  
`to_dict()` (nipyapi.nifi.models.state\_entry\_dto.StateEntryDTO method), 451  
`to_dict()` (nipyapi.nifi.models.state\_map\_dto.StateMapDTO method), 452  
`to_dict()` (nipyapi.nifi.models.status\_descriptor\_dto.StatusDescriptorDTO method), 453  
`to_dict()` (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO method), 454  
`to_dict()` (nipyapi.nifi.models.status\_history\_entity.StatusHistoryEntity method), 455  
`to_dict()` (nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO method), 456  
`to_dict()` (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO method), 457  
`to_dict()` (nipyapi.nifi.models.streaming\_output.StreamingOutput method), 458  
`to_dict()` (nipyapi.nifi.models.submit\_replay\_request\_entity.SubmitReplayRequestEntity method), 458  
`to_dict()` (nipyapi.nifi.models.system\_diagnostics\_dto.SystemDiagnosticsDTO method), 459  
`to_dict()` (nipyapi.nifi.models.system\_diagnostics\_entity.SystemDiagnosticsEntity method), 460  
`to_dict()` (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO method), 463  
`to_dict()` (nipyapi.nifi.models.template\_dto.TemplateDTO method), 466  
`to_dict()` (nipyapi.nifi.models.template\_entity.TemplateEntity method), 467  
`to_dict()` (nipyapi.nifi.models.templates\_entity.TemplatesEntity method), 468  
`to_dict()` (nipyapi.nifi.models.tenant\_dto.TenantDTO method), 469  
`to_dict()` (nipyapi.nifi.models.tenant\_entity.TenantEntity method), 471  
`to_dict()` (nipyapi.nifi.models.tenants\_entity.TenantsEntity method), 471  
`to_dict()` (nipyapi.nifi.models.transaction\_result\_entity.TransactionResultEntity method), 472  
`to_dict()` (nipyapi.nifi.models.update\_controller\_service\_reference\_request\_entity.UpdateControllerServiceReferenceRequestEntity method), 473  
`to_dict()` (nipyapi.nifi.models.user\_dto.UserDTO method), 475  
`to_dict()` (nipyapi.nifi.models.user\_entity.UserEntity method), 476  
`to_dict()` (nipyapi.nifi.models.user\_group\_dto.UserGroupDTO method), 477  
`to_dict()` (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity method), 479  
`to_dict()` (nipyapi.nifi.models.user\_groups\_entity.UserGroupsEntity method), 480  
`to_dict()` (nipyapi.nifi.models.users\_entity.UsersEntity method), 480  
`to_dict()` (nipyapi.nifi.models.variable\_dto.VariableDTO method), 481  
`to_dict()` (nipyapi.nifi.models.variable\_entity.VariableEntity method), 482  
`to_dict()` (nipyapi.nifi.models.variable\_registry\_dto.VariableRegistryDTO method), 483  
`to_dict()` (nipyapi.nifi.models.variable\_registry\_entity.VariableRegistryEntity method), 483  
`to_dict()` (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO method), 485  
`to_dict()` (nipyapi.nifi.models.variable\_registry\_update\_request\_entity.VariableRegistryUpdateRequestEntity method), 486  
`to_dict()` (nipyapi.nifi.models.variable\_registry\_update\_step\_dto.VariableRegistryUpdateStepDTO method), 487  
`to_dict()` (nipyapi.nifi.models.version\_control\_component\_mapping\_entity.VersionControlComponentMappingEntity method), 488  
`to_dict()` (nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO method), 490  
`to_dict()` (nipyapi.nifi.models.version\_control\_information\_entity.VersionControlInformationEntity method), 491  
`to_dict()` (nipyapi.nifi.models.version\_info\_dto.VersionInfoDTO method), 493  
`to_dict()` (nipyapi.nifi.models.versioned\_connection.VersionedConnection method), 496  
`to_dict()` (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService method), 499  
`to_dict()` (nipyapi.nifi.models.versioned\_flow.VersionedFlow method), 501



`to_dict()` (`nipyapi.nifi.models.versioned_flow_coordinates.VersionedFlowCoordinates`, `nipyapi.registry.models.fields.Fields` method), 502  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO`, `nipyapi.registry.models.permissions.Permissions` method), 504  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_entity.VersionedFlowEntity`, `nipyapi.registry.models.resource.Resource` method), 504  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_snapshot.VersionedFlowSnapshot`, `nipyapi.registry.models.resource_permissions.ResourcePermissions` method), 506  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity`, `nipyapi.registry.models.tenant.Tenant` method), 507  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity`, `nipyapi.registry.models.usage.Usage` method), 508  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity`, `nipyapi.registry.models.usage.Usage` method), 508  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_update_request_dto.VersionedFlowUpdateRequestDTO`, `nipyapi.registry.models.versioned_flow_update_request_dto.VersionedFlowUpdateRequestDTO` method), 510  
`to_dict()` (`nipyapi.nifi.models.versioned_flow_update_request_entity.VersionedFlowUpdateRequestEntity`, `nipyapi.registry.models.versioned_flow_update_request_entity.VersionedFlowUpdateRequestEntity` method), 511  
`to_dict()` (`nipyapi.nifi.models.versioned_flows_entity.VersionedFlowsEntity`, `nipyapi.registry.models.versioned_flow.VersionedFlow` method), 511  
`to_dict()` (`nipyapi.nifi.models.versioned_funnel.VersionedFunnel`, `nipyapi.registry.models.versioned_flow_coordinates.VersionedFlowCoordinates` method), 513  
`to_dict()` (`nipyapi.nifi.models.versioned_label.VersionedLabel`, `nipyapi.registry.models.versioned_flow_snapshot.VersionedFlowSnapshot` method), 514  
`to_dict()` (`nipyapi.nifi.models.versioned_port.VersionedPort`, `nipyapi.registry.models.versioned_flow_snapshot_metadata.VersionedFlowSnapshotMetadata` method), 516  
`to_dict()` (`nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup`, `nipyapi.registry.models.versioned_funnel.VersionedFunnel` method), 521  
`to_dict()` (`nipyapi.nifi.models.versioned_processor.VersionedProcessor`, `nipyapi.registry.models.versioned_label.VersionedLabel` method), 525  
`to_dict()` (`nipyapi.nifi.models.versioned_property_descriptor.VersionedPropertyDescriptor`, `nipyapi.registry.models.versioned_port.VersionedPort` method), 527  
`to_dict()` (`nipyapi.nifi.models.versioned_remote_group_port.VersionedRemoteGroupPort`, `nipyapi.registry.models.versioned_process_group.VersionedProcessGroup` method), 529  
`to_dict()` (`nipyapi.nifi.models.versioned_remote_process_group.VersionedRemoteProcessGroup`, `nipyapi.registry.models.versioned_processor.VersionedProcessor` method), 532  
`to_dict()` (`nipyapi.registry.models.access_policy.AccessPolicy`, `nipyapi.registry.models.versioned_property_descriptor.VersionedPropertyDescriptor` method), 569  
`to_dict()` (`nipyapi.registry.models.access_policy_summary.AccessPolicySummary`, `nipyapi.registry.models.versioned_remote_group_port.VersionedRemoteGroupPort` method), 570  
`to_dict()` (`nipyapi.registry.models.batch_size.BatchSize`, `nipyapi.registry.models.versioned_remote_process_group.VersionedRemoteProcessGroup` method), 571  
`to_dict()` (`nipyapi.registry.models.bucket.Bucket`, `nipyapi.nifi.models.about_dto.AboutDTO` method), 573  
`to_dict()` (`nipyapi.registry.models.bucket_item.BucketItem`, `nipyapi.nifi.models.about_entity.AboutEntity` method), 574  
`to_dict()` (`nipyapi.registry.models.bundle.Bundle`, `nipyapi.nifi.models.access_configuration_dto.AccessConfigurationDTO` method), 575  
`to_dict()` (`nipyapi.registry.models.connectable_component.ConnectionComponent`, `nipyapi.nifi.models.access_configuration_entity.AccessConfigurationEntity` method), 576  
`to_dict()` (`nipyapi.registry.models.controller_service_api.ControllerServiceAPI`, `nipyapi.nifi.models.access_policy_dto.AccessPolicyDTO` method), 577  
`to_dict()` (`nipyapi.registry.models.current_user.CurrentUser`, `nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity` method), 578

`to_str()` (`nipyapi.nifi.models.access_policy_summary_dto.AccessPolicySummaryDTO` method), 198  
`to_str()` (`nipyapi.nifi.models.access_policy_summary_entity.AccessPolicySummaryEntity` method), 198  
`to_str()` (`nipyapi.nifi.models.access_status_dto.AccessStatusDTO` method), 201  
`to_str()` (`nipyapi.nifi.models.access_status_entity.AccessStatusEntity` method), 202  
`to_str()` (`nipyapi.nifi.models.action_details_dto.ActionDetailsDTO` method), 202  
`to_str()` (`nipyapi.nifi.models.action_dto.ActionDTO` method), 203  
`to_str()` (`nipyapi.nifi.models.action_entity.ActionEntity` method), 204  
`to_str()` (`nipyapi.nifi.models.activate_controller_services_entity.ActivateControllerServicesEntity` method), 205  
`to_str()` (`nipyapi.nifi.models.affected_component_dto.AffectedComponentDTO` method), 207  
`to_str()` (`nipyapi.nifi.models.affected_component_entity.AffectedComponentEntity` method), 209  
`to_str()` (`nipyapi.nifi.models.allowable_value_dto.AllowableValueDTO` method), 210  
`to_str()` (`nipyapi.nifi.models.allowable_value_entity.AllowableValueEntity` method), 210  
`to_str()` (`nipyapi.nifi.models.attribute_dto.AttributeDTO` method), 211  
`to_str()` (`nipyapi.nifi.models.banner_dto.BannerDTO` method), 212  
`to_str()` (`nipyapi.nifi.models.banner_entity.BannerEntity` method), 212  
`to_str()` (`nipyapi.nifi.models.batch_settings_dto.BatchSettingsDTO` method), 213  
`to_str()` (`nipyapi.nifi.models.batch_size.BatchSize` method), 214  
`to_str()` (`nipyapi.nifi.models.bucket.Bucket` method), 215  
`to_str()` (`nipyapi.nifi.models.bucket_dto.BucketDTO` method), 216  
`to_str()` (`nipyapi.nifi.models.bucket_entity.BucketEntity` method), 217  
`to_str()` (`nipyapi.nifi.models.buckets_entity.BucketsEntity` method), 218  
`to_str()` (`nipyapi.nifi.models.bulletin_board_dto.BulletinBoardDTO` method), 218  
`to_str()` (`nipyapi.nifi.models.bulletin_board_entity.BulletinBoardEntity` method), 219  
`to_str()` (`nipyapi.nifi.models.bulletin_dto.BulletinDTO` method), 220  
`to_str()` (`nipyapi.nifi.models.bulletin_entity.BulletinEntity` method), 222  
`to_str()` (`nipyapi.nifi.models.bundle.Bundle` method), 222  
`to_str()` (`nipyapi.nifi.models.bundle_dto.BundleDTO` method), 223  
`to_str()` (`nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO` method), 224  
`to_str()` (`nipyapi.nifi.models.cluster_summary_entity.ClusterSummaryEntity` method), 224  
`to_str()` (`nipyapi.nifi.models.cluster_dto.ClusterDTO` method), 224  
`to_str()` (`nipyapi.nifi.models.cluster_entity.ClusterEntity` method), 225  
`to_str()` (`nipyapi.nifi.models.cluster_search_results_entity.ClusterSearchResultsEntity` method), 226  
`to_str()` (`nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO` method), 227  
`to_str()` (`nipyapi.nifi.models.component_details_dto.ComponentDetailsDTO` method), 227  
`to_str()` (`nipyapi.nifi.models.component_difference_dto.ComponentDifferenceDTO` method), 228  
`to_str()` (`nipyapi.nifi.models.component_history_dto.ComponentHistoryDTO` method), 229  
`to_str()` (`nipyapi.nifi.models.component_history_entity.ComponentHistoryEntity` method), 230  
`to_str()` (`nipyapi.nifi.models.component_reference_dto.ComponentReferenceDTO` method), 231  
`to_str()` (`nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity` method), 232  
`to_str()` (`nipyapi.nifi.models.component_search_result_dto.ComponentSearchResultDTO` method), 234  
`to_str()` (`nipyapi.nifi.models.component_state_dto.ComponentStateDTO` method), 235  
`to_str()` (`nipyapi.nifi.models.component_state_entity.ComponentStateEntity` method), 235  
`to_str()` (`nipyapi.nifi.models.connectable_component.ConnectableComponent` method), 237  
`to_str()` (`nipyapi.nifi.models.connectable_dto.ConnectableDTO` method), 238  
`to_str()` (`nipyapi.nifi.models.connection_dto.ConnectionDTO` method), 241  
`to_str()` (`nipyapi.nifi.models.connection_entity.ConnectionEntity` method), 244  
`to_str()` (`nipyapi.nifi.models.connection_status_dto.ConnectionStatusDTO` method), 246  
`to_str()` (`nipyapi.nifi.models.connection_status_entity.ConnectionStatusEntity` method), 247  
`to_str()` (`nipyapi.nifi.models.connection_status_snapshot_dto.ConnectionStatusSnapshotDTO` method), 251  
`to_str()` (`nipyapi.nifi.models.connection_status_snapshot_entity.ConnectionStatusSnapshotEntity` method), 252  
`to_str()` (`nipyapi.nifi.models.connections_entity.ConnectionsEntity` method), 252  
`to_str()` (`nipyapi.nifi.models.controller_bulletins_entity.ControllerBulletinsEntity` method), 253  
`to_str()` (`nipyapi.nifi.models.controller_configuration_dto.ControllerConfigurationDTO` method), 254  
`to_str()` (`nipyapi.nifi.models.controller_configuration_entity.ControllerConfigurationEntity` method), 255  
`to_str()` (`nipyapi.nifi.models.controller_dto.ControllerDTO` method), 258

`to_str()` (`nipyapi.nifi.models.controller_entity.ControllerEntity` to\_str method), 258  
`to_str()` (`nipyapi.nifi.models.controller_service_api.ControllerServiceAPI` to\_str method), 259  
`to_str()` (`nipyapi.nifi.models.controller_service_api_dto.ControllerServiceAPIDTO` to\_str method), 260  
`to_str()` (`nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO` to\_str method), 264  
`to_str()` (`nipyapi.nifi.models.controller_service_entity.ControllerServiceEntity` to\_str method), 266  
`to_str()` (`nipyapi.nifi.models.controller_service_referencing_component_entity.ControllerServiceReferencingComponentEntity` to\_str method), 269  
`to_str()` (`nipyapi.nifi.models.controller_service_referencing_component_entity_dto.ControllerServiceReferencingComponentEntityDTO` to\_str method), 271  
`to_str()` (`nipyapi.nifi.models.controller_service_referencing_component_entity_dto.ControllerServiceReferencingComponentEntityDTO` to\_str method), 272  
`to_str()` (`nipyapi.nifi.models.controller_service_types_entity.ControllerServiceTypesEntity` to\_str method), 272  
`to_str()` (`nipyapi.nifi.models.controller_services_entity.ControllerServicesEntity` to\_str method), 273  
`to_str()` (`nipyapi.nifi.models.controller_status_dto.ControllerStatusDTO` to\_str method), 276  
`to_str()` (`nipyapi.nifi.models.controller_status_entity.ControllerStatusEntity` to\_str method), 276  
`to_str()` (`nipyapi.nifi.models.copy_snippet_request_entity.CopySnippetRequestEntity` to\_str method), 278  
`to_str()` (`nipyapi.nifi.models.counter_dto.CounterDTO` to\_str method), 278  
`to_str()` (`nipyapi.nifi.models.counter_entity.CounterEntity` to\_str method), 279  
`to_str()` (`nipyapi.nifi.models.counters_dto.CountersDTO` to\_str method), 280  
`to_str()` (`nipyapi.nifi.models.counters_entity.CountersEntity` to\_str method), 280  
`to_str()` (`nipyapi.nifi.models.counters_snapshot_dto.CountersSnapshotDTO` to\_str method), 281  
`to_str()` (`nipyapi.nifi.models.create_active_request_entity.CreateActiveRequestEntity` to\_str method), 282  
`to_str()` (`nipyapi.nifi.models.create_template_request_entity.CreateTemplateRequestEntity` to\_str method), 283  
`to_str()` (`nipyapi.nifi.models.current_user_entity.CurrentUserEntity` to\_str method), 285  
`to_str()` (`nipyapi.nifi.models.difference_dto.DifferenceDTO` to\_str method), 286  
`to_str()` (`nipyapi.nifi.models.dimensions_dto.DimensionsDTO` to\_str method), 287  
`to_str()` (`nipyapi.nifi.models.documented_type_dto.DocumentedTypeDTO` to\_str method), 288  
`to_str()` (`nipyapi.nifi.models.drop_request_dto.DropRequestDTO` to\_str method), 291  
`to_str()` (`nipyapi.nifi.models.drop_request_entity.DropRequestEntity` to\_str method), 292  
`to_str()` (`nipyapi.nifi.models.flow_breadcrumb_dto.FlowBreadcrumbDTO` to\_str method), 292  
`to_str()` (`nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity` to\_str method), 293  
`to_str()` (`nipyapi.nifi.models.flow_comparison_entity.FlowComparisonEntity` to\_str method), 294  
`to_str()` (`nipyapi.nifi.models.flow_configuration_dto.FlowConfigurationDTO` to\_str method), 296  
`to_str()` (`nipyapi.nifi.models.flow_configuration_entity.FlowConfigurationEntity` to\_str method), 296  
`to_str()` (`nipyapi.nifi.models.flow_dto.FlowDTO` to\_str method), 298  
`to_str()` (`nipyapi.nifi.models.flow_entity.FlowEntity` to\_str method), 298  
`to_str()` (`nipyapi.nifi.models.flow_file_reference_entity.FlowFileReferenceEntity` to\_str method), 301  
`to_str()` (`nipyapi.nifi.models.flow_file_summary_dto.FlowFileSummaryDTO` to\_str method), 303  
`to_str()` (`nipyapi.nifi.models.flow_snippet_dto.FlowSnippetDTO` to\_str method), 305  
`to_str()` (`nipyapi.nifi.models.funnel_dto.FunnelDTO` to\_str method), 306  
`to_str()` (`nipyapi.nifi.models.funnel_entity.FunnelEntity` to\_str method), 307  
`to_str()` (`nipyapi.nifi.models.funnels_entity.FunnelsEntity` to\_str method), 308  
`to_str()` (`nipyapi.nifi.models.garbage_collection_dto.GarbageCollectionDTO` to\_str method), 309  
`to_str()` (`nipyapi.nifi.models.history_dto.HistoryDTO` to\_str method), 310  
`to_str()` (`nipyapi.nifi.models.history_entity.HistoryEntity` to\_str method), 310  
`to_str()` (`nipyapi.nifi.models.input_ports_entity.InputPortsEntity` to\_str method), 311  
`to_str()` (`nipyapi.nifi.models.instantiate_template_request_entity.InstantiateTemplateRequestEntity` to\_str method), 312  
`to_str()` (`nipyapi.nifi.models.label_dto.LabelDTO` to\_str method), 314  
`to_str()` (`nipyapi.nifi.models.label_entity.LabelEntity` to\_str method), 315  
`to_str()` (`nipyapi.nifi.models.labels_entity.LabelsEntity` to\_str method), 316  
`to_str()` (`nipyapi.nifi.models.lineage_dto.LineageDTO` to\_str method), 317  
`to_str()` (`nipyapi.nifi.models.lineage_entity.LineageEntity` to\_str method), 318  
`to_str()` (`nipyapi.nifi.models.lineage_request_dto.LineageRequestDTO` to\_str method), 319  
`to_str()` (`nipyapi.nifi.models.lineage_results_dto.LineageResultsDTO` to\_str method), 320  
`to_str()` (`nipyapi.nifi.models.listing_request_dto.ListingRequestDTO` to\_str method), 322  
`to_str()` (`nipyapi.nifi.models.listing_request_entity.ListingRequestEntity` to\_str method), 322

<code>to_str()</code> (nipyapi.nifi.models.node_connection_status_snapshot_dto.NodeConnectionStatusSnapshotDTO method), 323	<code>to_str()</code> (nipyapi.nifi.models.node_status_snapshot_dto.NodeStatusSnapshotDTO method), 360
<code>to_str()</code> (nipyapi.nifi.models.node_counters_snapshot_dto.NodeCountersSnapshotDTO method), 324	<code>to_str()</code> (nipyapi.nifi.models.process_group_flow_dto.ProcessGroupFlowDTO method), 361
<code>to_str()</code> (nipyapi.nifi.models.node_dto.NodeDTO method), 326	<code>to_str()</code> (nipyapi.nifi.models.process_group_flow_entity.ProcessGroupFlowEntity method), 362
<code>to_str()</code> (nipyapi.nifi.models.node_entity.NodeEntity method), 327	<code>to_str()</code> (nipyapi.nifi.models.process_group_status_dto.ProcessGroupStatusDTO method), 363
<code>to_str()</code> (nipyapi.nifi.models.node_event_dto.NodeEventDTO method), 327	<code>to_str()</code> (nipyapi.nifi.models.process_group_status_entity.ProcessGroupStatusEntity method), 364
<code>to_str()</code> (nipyapi.nifi.models.node_port_status_snapshot_dto.NodePortStatusSnapshotDTO method), 328	<code>to_str()</code> (nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDTO method), 369
<code>to_str()</code> (nipyapi.nifi.models.node_process_group_status_snapshot_dto.NodeProcessGroupStatusSnapshotDTO method), 329	<code>to_str()</code> (nipyapi.nifi.models.process_group_status_snapshot_entity.ProcessGroupStatusSnapshotEntity method), 371
<code>to_str()</code> (nipyapi.nifi.models.node_processor_status_snapshot_dto.NodeProcessorStatusSnapshotDTO method), 330	<code>to_str()</code> (nipyapi.nifi.models.process_group_status_snapshot_entity.ProcessGroupStatusSnapshotEntity method), 371
<code>to_str()</code> (nipyapi.nifi.models.node_remote_process_group_status_snapshot_dto.NodeRemoteProcessGroupStatusSnapshotDTO method), 331	<code>to_str()</code> (nipyapi.nifi.models.processor_config_status_snapshot_dto.ProcessorConfigStatusSnapshotDTO method), 375
<code>to_str()</code> (nipyapi.nifi.models.node_search_result_dto.NodeSearchResultDTO method), 332	<code>to_str()</code> (nipyapi.nifi.models.processor_dto.ProcessorDTO method), 378
<code>to_str()</code> (nipyapi.nifi.models.node_status_snapshots_dto.NodeStatusSnapshotsDTO method), 333	<code>to_str()</code> (nipyapi.nifi.models.processor_entity.ProcessorEntity method), 381
<code>to_str()</code> (nipyapi.nifi.models.node_system_diagnostics_snapshot_dto.NodeSystemDiagnosticsSnapshotDTO method), 334	<code>to_str()</code> (nipyapi.nifi.models.processor_status_dto.ProcessorStatusDTO method), 382
<code>to_str()</code> (nipyapi.nifi.models.output_ports_entity.OutputPortsEntity method), 335	<code>to_str()</code> (nipyapi.nifi.models.processor_status_entity.ProcessorStatusEntity method), 383
<code>to_str()</code> (nipyapi.nifi.models.peer_dto.PeerDTO method), 335	<code>to_str()</code> (nipyapi.nifi.models.processor_status_snapshot_dto.ProcessorStatusSnapshotDTO method), 387
<code>to_str()</code> (nipyapi.nifi.models.peers_entity.PeersEntity method), 336	<code>to_str()</code> (nipyapi.nifi.models.processor_status_snapshot_entity.ProcessorStatusSnapshotEntity method), 388
<code>to_str()</code> (nipyapi.nifi.models.permissions.Permissions method), 337	<code>to_str()</code> (nipyapi.nifi.models.processor_types_entity.ProcessorTypesEntity method), 388
<code>to_str()</code> (nipyapi.nifi.models.permissions_dto.PermissionsDTO method), 337	<code>to_str()</code> (nipyapi.nifi.models.processors_entity.ProcessorsEntity method), 389
<code>to_str()</code> (nipyapi.nifi.models.port_dto.PortDTO method), 339	<code>to_str()</code> (nipyapi.nifi.models.property_descriptor_dto.PropertyDescriptorDTO method), 392
<code>to_str()</code> (nipyapi.nifi.models.port_entity.PortEntity method), 341	<code>to_str()</code> (nipyapi.nifi.models.property_descriptor_entity.PropertyDescriptorEntity method), 392
<code>to_str()</code> (nipyapi.nifi.models.port_status_dto.PortStatusDTO method), 343	<code>to_str()</code> (nipyapi.nifi.models.property_history_dto.PropertyHistoryDTO method), 393
<code>to_str()</code> (nipyapi.nifi.models.port_status_entity.PortStatusEntity method), 344	<code>to_str()</code> (nipyapi.nifi.models.provenance_dto.ProvenanceDTO method), 394
<code>to_str()</code> (nipyapi.nifi.models.port_status_snapshot_dto.PortStatusSnapshotDTO method), 346	<code>to_str()</code> (nipyapi.nifi.models.provenance_entity.ProvenanceEntity method), 395
<code>to_str()</code> (nipyapi.nifi.models.port_status_snapshot_entity.PortStatusSnapshotEntity method), 346	<code>to_str()</code> (nipyapi.nifi.models.provenance_event_dto.ProvenanceEventDTO method), 401
<code>to_str()</code> (nipyapi.nifi.models.position_dto.PositionDTO method), 347	<code>to_str()</code> (nipyapi.nifi.models.provenance_event_entity.ProvenanceEventEntity method), 402
<code>to_str()</code> (nipyapi.nifi.models.previous_value_dto.PreviousValueDTO method), 348	<code>to_str()</code> (nipyapi.nifi.models.provenance_link_dto.ProvenanceLinkDTO method), 403
<code>to_str()</code> (nipyapi.nifi.models.prioritizer_types_entity.PrioritizerTypesEntity method), 348	<code>to_str()</code> (nipyapi.nifi.models.provenance_node_dto.ProvenanceNodeDTO method), 405
<code>to_str()</code> (nipyapi.nifi.models.process_group_dto.ProcessGroupDTO method), 354	<code>to_str()</code> (nipyapi.nifi.models.provenance_options_dto.ProvenanceOptionsDTO method), 405



`to_str()` (nipyapi.nifi.models.provenance\_options\_entity.ProvenanceOptionsEntity method), 406  
`to_str()` (nipyapi.nifi.models.provenance\_request\_dto.ProvenanceRequestDTO method), 408  
`to_str()` (nipyapi.nifi.models.provenance\_results\_dto.ProvenanceResultsDTO method), 409  
`to_str()` (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO method), 410  
`to_str()` (nipyapi.nifi.models.queue\_size\_dto.QueueSizeDTO method), 411  
`to_str()` (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity method), 413  
`to_str()` (nipyapi.nifi.models.registry\_clients\_entity.RegistryClientsEntity method), 413  
`to_str()` (nipyapi.nifi.models.registry\_dto.RegistryDTO method), 414  
`to_str()` (nipyapi.nifi.models.relationship\_dto.RelationshipDTO method), 415  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_contents\_dto.RemoteProcessGroupContentsDTO method), 416  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_dto.RemoteProcessGroupDTO method), 420  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity method), 423  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_port\_dto.RemoteProcessGroupPortDTO method), 425  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity method), 427  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_status\_dto.RemoteProcessGroupStatusDTO method), 429  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_status\_entity.RemoteProcessGroupStatusEntity method), 430  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_status\_snapshots\_dto.RemoteProcessGroupStatusSnapshotsDTO method), 432  
`to_str()` (nipyapi.nifi.models.remote\_process\_group\_status\_snapshots\_entity.RemoteProcessGroupStatusSnapshotsEntity method), 433  
`to_str()` (nipyapi.nifi.models.remote\_process\_groups\_entity.RemoteProcessGroupsEntity method), 434  
`to_str()` (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO method), 438  
`to_str()` (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity method), 440  
`to_str()` (nipyapi.nifi.models.reporting\_task\_types\_entity.ReportingTaskTypesEntity method), 441  
`to_str()` (nipyapi.nifi.models.reporting\_tasks\_entity.ReportingTasksEntity method), 441  
`to_str()` (nipyapi.nifi.models.resource\_dto.ResourceDTO method), 442  
`to_str()` (nipyapi.nifi.models.resources\_entity.ResourcesEntity method), 442  
`to_str()` (nipyapi.nifi.models.revision\_dto.RevisionDTO method), 443  
`to_str()` (nipyapi.nifi.models.schedule\_components\_entity.ScheduleComponentsEntity method), 444  
`to_str()` (nipyapi.nifi.models.search\_results\_dto.SearchResultsDTO method), 446  
`to_str()` (nipyapi.nifi.models.search\_results\_entity.SearchResultsEntity method), 447  
`to_str()` (nipyapi.nifi.models.snippet\_dto.SnippetDTO method), 449  
`to_str()` (nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity method), 450  
`to_str()` (nipyapi.nifi.models.state\_entry\_dto.StateEntryDTO method), 451  
`to_str()` (nipyapi.nifi.models.state\_map\_dto.StateMapDTO method), 452  
`to_str()` (nipyapi.nifi.models.status\_descriptor\_dto.StatusDescriptorDTO method), 453  
`to_str()` (nipyapi.nifi.models.status\_history\_dto.StatusHistoryDTO method), 454  
`to_str()` (nipyapi.nifi.models.status\_history\_entity.StatusHistoryEntity method), 455  
`to_str()` (nipyapi.nifi.models.status\_snapshot\_dto.StatusSnapshotDTO method), 456  
`to_str()` (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO method), 457  
`to_str()` (nipyapi.nifi.models.streaming\_output\_dto.StreamingOutputDTO method), 458  
`to_str()` (nipyapi.nifi.models.submit\_replay\_request\_entity.SubmitReplayRequestEntity method), 458  
`to_str()` (nipyapi.nifi.models.system\_diagnostics\_dto.SystemDiagnosticsDTO method), 459  
`to_str()` (nipyapi.nifi.models.system\_diagnostics\_entity.SystemDiagnosticsEntity method), 460  
`to_str()` (nipyapi.nifi.models.system\_snapshot\_dto.SystemSnapshotDTO method), 463  
`to_str()` (nipyapi.nifi.models.system\_snapshot\_entity.SystemSnapshotEntity method), 466  
`to_str()` (nipyapi.nifi.models.template\_entity.TemplateEntity method), 467  
`to_str()` (nipyapi.nifi.models.templates\_entity.TemplatesEntity method), 468  
`to_str()` (nipyapi.nifi.models.tenant\_dto.TenantDTO method), 469  
`to_str()` (nipyapi.nifi.models.tenant\_entity.TenantEntity method), 471  
`to_str()` (nipyapi.nifi.models.tenants\_entity.TenantsEntity method), 471  
`to_str()` (nipyapi.nifi.models.transaction\_result\_entity.TransactionResultEntity method), 472  
`to_str()` (nipyapi.nifi.models.update\_controller\_service\_reference\_request\_entity.UpdateControllerServiceReferenceRequestEntity method), 473  
`to_str()` (nipyapi.nifi.models.user\_dto.UserDTO method), 475  
`to_str()` (nipyapi.nifi.models.user\_entity.UserEntity method), 476

`to_str()` (`nipyapi.nifi.models.user_group_dto.UserGroupDTO` to\_str method), 478  
`to_str()` (`nipyapi.nifi.models.user_group_entity.UserGroupEntity` to\_str method), 479  
`to_str()` (`nipyapi.nifi.models.user_groups_entity.UserGroupsEntity` to\_str method), 480  
`to_str()` (`nipyapi.nifi.models.users_entity.UsersEntity` to\_str method), 480  
`to_str()` (`nipyapi.nifi.models.variable_dto.VariableDTO` to\_str method), 481  
`to_str()` (`nipyapi.nifi.models.variable_entity.VariableEntity` to\_str method), 482  
`to_str()` (`nipyapi.nifi.models.variable_registry_dto.VariableRegistryDTO` to\_str method), 483  
`to_str()` (`nipyapi.nifi.models.variable_registry_entity.VariableRegistryEntity` to\_str method), 484  
`to_str()` (`nipyapi.nifi.models.variable_registry_update_request_dto.VariableRegistryUpdateRequestDTO` to\_str method), 485  
`to_str()` (`nipyapi.nifi.models.variable_registry_update_request_entity.VariableRegistryUpdateRequestEntity` to\_str method), 486  
`to_str()` (`nipyapi.nifi.models.variable_registry_update_step_dto.VariableRegistryUpdateStepDTO` to\_str method), 487  
`to_str()` (`nipyapi.nifi.models.version_control_component_mapping_entity.VersionControlComponentMappingEntity` to\_str method), 488  
`to_str()` (`nipyapi.nifi.models.version_control_information_dto.VersionControlInformationDTO` to\_str method), 490  
`to_str()` (`nipyapi.nifi.models.version_control_information_entity.VersionControlInformationEntity` to\_str method), 491  
`to_str()` (`nipyapi.nifi.models.version_info_dto.VersionInfoDTO` to\_str method), 493  
`to_str()` (`nipyapi.nifi.models.versioned_connection.VersionedConnection` to\_str method), 497  
`to_str()` (`nipyapi.nifi.models.versioned_controller_service.VersionedControllerService` to\_str method), 499  
`to_str()` (`nipyapi.nifi.models.versioned_flow.VersionedFlow` to\_str method), 501  
`to_str()` (`nipyapi.nifi.models.versioned_flow_coordinates.VersionedFlowCoordinates` to\_str method), 502  
`to_str()` (`nipyapi.nifi.models.versioned_flow_dto.VersionedFlowDTO` to\_str method), 504  
`to_str()` (`nipyapi.nifi.models.versioned_flow_entity.VersionedFlowEntity` to\_str method), 504  
`to_str()` (`nipyapi.nifi.models.versioned_flow_snapshot.VersionedFlowSnapshot` to\_str method), 506  
`to_str()` (`nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity` to\_str method), 507  
`to_str()` (`nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity` to\_str method), 508  
`to_str()` (`nipyapi.nifi.models.versioned_flow_snapshot_metadata_set_entity.VersionedFlowSnapshotMetadataSetEntity` to\_str method), 508  
`to_str()` (`nipyapi.nifi.models.versioned_flow_update_requests_dto.VersionedFlowUpdateRequestsDTO` to\_str method), 510  
`to_str()` (`nipyapi.nifi.models.versioned_flow_update_requests_entity.VersionedFlowUpdateRequestsEntity` to\_str method), 511  
`to_str()` (`nipyapi.nifi.models.versioned_flows_entity.VersionedFlowsEntity` to\_str method), 511  
`to_str()` (`nipyapi.nifi.models.versioned_funnel.VersionedFunnel` to\_str method), 513  
`to_str()` (`nipyapi.nifi.models.versioned_label.VersionedLabel` to\_str method), 514  
`to_str()` (`nipyapi.nifi.models.versioned_port.VersionedPort` to\_str method), 516  
`to_str()` (`nipyapi.nifi.models.versioned_process_group.VersionedProcessGroup` to\_str method), 521  
`to_str()` (`nipyapi.nifi.models.versioned_processor.VersionedProcessor` to\_str method), 525  
`to_str()` (`nipyapi.nifi.models.versioned_property_descriptor.VersionedPropertyDescriptor` to\_str method), 527  
`to_str()` (`nipyapi.nifi.models.versioned_remote_group_port.VersionedRemoteGroupPort` to\_str method), 529  
`to_str()` (`nipyapi.nifi.models.versioned_update_request_entity.VersionedUpdateRequestEntity` to\_str method), 532  
`to_str()` (`nipyapi.nifi.models.versioned_access_policy.VersionedAccessPolicy` to\_str method), 569  
`to_str()` (`nipyapi.nifi.models.bucket.Bucket` to\_str method), 573  
`to_str()` (`nipyapi.nifi.models.bucket_item.BucketItem` to\_str method), 574  
`to_str()` (`nipyapi.registry.models.bundle.Bundle` to\_str method), 575  
`to_str()` (`nipyapi.registry.models.connectable_component.ConnectableComponent` to\_str method), 576  
`to_str()` (`nipyapi.registry.models.controller_service_api.ControllerServiceApi` to\_str method), 577  
`to_str()` (`nipyapi.registry.models.current_user.CurrentUser` to\_str method), 578  
`to_str()` (`nipyapi.registry.models.fields.Fields` to\_str method), 579  
`to_str()` (`nipyapi.registry.models.permissions.Permissions` to\_str method), 579  
`to_str()` (`nipyapi.registry.models.resource.Resource` to\_str method), 580  
`to_str()` (`nipyapi.registry.models.resource_permissions.ResourcePermissions` to\_str method), 581  
`to_str()` (`nipyapi.registry.models.tenant.Tenant` to\_str method), 582  
`to_str()` (`nipyapi.registry.models.tenant_group.TenantGroup` to\_str method), 584  
`to_str()` (`nipyapi.registry.models.versioned_flow_snapshot_metadata_set_entity.VersionedFlowSnapshotMetadataSetEntity` to\_str method), 585  
`to_str()` (`nipyapi.registry.models.versioned_connection.VersionedConnection` to\_str method), 589  
`to_str()` (`nipyapi.registry.models.versioned_controller_service.VersionedControllerService` to\_str method), 592

`to_str()` (`nipyapi.registry.models.versioned_flow.VersionedFlow` attribute), 594  
`transfer_flow_files()` (`nipyapi.nifi.apis.data_transfer_api.DataTransferApi` method), 594  
`to_str()` (`nipyapi.registry.models.versioned_flow_coordinates.VersionedFlowCoordinates` attribute), 595  
`transfer_flow_files_with_http_info()` (`nipyapi.nifi.apis.data_transfer_api.DataTransferApi` method), 597  
`to_str()` (`nipyapi.registry.models.versioned_flow_snapshot.VersionedFlowSnapshot` attribute), 597  
`to_str()` (`nipyapi.registry.models.versioned_flow_snapshot_metadata.VersionedFlowSnapshotMetadata` attribute), 598  
`to_str()` (`nipyapi.registry.models.versioned_funnel.VersionedFunnel` attribute), 599  
`to_str()` (`nipyapi.registry.models.versioned_label.VersionedLabel` attribute), 601  
`to_str()` (`nipyapi.registry.models.versioned_port.VersionedPort` attribute), 603  
`to_str()` (`nipyapi.registry.models.versioned_process_group.VersionedProcessGroup` attribute), 607  
`to_str()` (`nipyapi.registry.models.versioned_processor.VersionedProcessor` attribute), 611  
`to_str()` (`nipyapi.registry.models.versioned_property_descriptor.VersionedPropertyDescriptor` attribute), 613  
`to_str()` (`nipyapi.registry.models.versioned_remote_group_port.VersionedRemoteGroupPort` attribute), 615  
`to_str()` (`nipyapi.registry.models.versioned_remote_process_group.VersionedRemoteProcessGroup` attribute), 618  
`total` (`nipyapi.nifi.models.history_dto.HistoryDTO` attribute), 310  
`total` (`nipyapi.nifi.models.provenance_results_dto.ProvenanceResultsDTO` attribute), 409  
`total_count` (`nipyapi.nifi.models.provenance_results_dto.ProvenanceResultsDTO` attribute), 409  
`total_entry_count` (`nipyapi.nifi.models.state_map_dto.StateMapDTO` attribute), 452  
`total_heap` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`total_heap_bytes` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`total_node_count` (`nipyapi.nifi.models.cluster_summary_dto.ClusterSummaryDTO` attribute), 227  
`total_non_heap` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`total_non_heap_bytes` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`total_space` (`nipyapi.nifi.models.storage_usage_dto.StorageUsageDTO` attribute), 457  
`total_space_bytes` (`nipyapi.nifi.models.storage_usage_dto.StorageUsageDTO` attribute), 457  
`total_threads` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`TransactionResultEntity` (class in `nipyapi.nifi.models.port_dto.PortDTO`), 472  
`transmitting` (`nipyapi.nifi.models.remote_process_group_dto.RemoteProcessGroupDTO` attribute), 420  
`transport_protocol` (`nipyapi.nifi.models.versioned_remote_process_group.VersionedRemoteProcessGroup` attribute), 421  
`transport_protocol` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDTO` attribute), 464  
`transport_protocol` (`nipyapi.nifi.models.versioned_remote_process_group.VersionedRemoteProcessGroup` attribute), 421  
`type` (`nipyapi.nifi.models.controller_service_api.ControllerServiceAPI` attribute), 260  
`type` (`nipyapi.nifi.models.controller_service_api_dto.ControllerServiceAPI` attribute), 264  
`type` (`nipyapi.nifi.models.controller_service_dto.ControllerServiceDTO` attribute), 269  
`type` (`nipyapi.nifi.models.controller_service_referencing_component_dto.ControllerServiceReferencingComponentDTO` attribute), 288  
`type` (`nipyapi.nifi.models.processor_dto.ProcessorDTO` attribute), 339

attribute), 379  
 type (nipyapi.nifi.models.processor\_status\_dto.ProcessorStatusDTO(nipyapi.registry.apis.policies\_api.PoliciesApi  
 attribute), 382  
 type (nipyapi.nifi.models.processor\_status\_snapshot\_dto.ProcessorStatusSnapshotDTO(nipyapi.registry.apis.buckets\_api.BucketsApi  
 attribute), 387  
 type (nipyapi.nifi.models.provenance\_node\_dto.ProvenanceNodeDTO(nipyapi.registry.apis.buckets\_api.BucketsApi  
 attribute), 405  
 type (nipyapi.nifi.models.provenance\_searchable\_field\_dto.ProvenanceSearchableFieldDTO  
 attribute), 410  
 type (nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO  
 attribute), 438  
 type (nipyapi.nifi.models.versioned\_controller\_service.VersionedControllerService  
 attribute), 499  
 type (nipyapi.nifi.models.versioned\_flow.VersionedFlow  
 attribute), 501  
 type (nipyapi.nifi.models.versioned\_port.VersionedPort  
 attribute), 516  
 type (nipyapi.nifi.models.versioned\_processor.VersionedProcessor  
 attribute), 525  
 type (nipyapi.registry.models.bucket\_item.BucketItem  
 attribute), 574  
 type (nipyapi.registry.models.connectable\_component.ConnectableComponent  
 attribute), 576  
 type (nipyapi.registry.models.controller\_service\_api.ControllerServiceApi  
 attribute), 577  
 type (nipyapi.registry.models.versioned\_controller\_service.VersionedControllerService  
 attribute), 592  
 type (nipyapi.registry.models.versioned\_flow.VersionedFlow  
 attribute), 594  
 type (nipyapi.registry.models.versioned\_port.VersionedPort  
 attribute), 603  
 type (nipyapi.registry.models.versioned\_processor.VersionedProcessor  
 attribute), 611

**U**

ui\_only (nipyapi.nifi.models.update\_controller\_service\_reference\_request\_entity.UpdateControllerServiceReferenceRequestEntity  
 attribute), 473  
 up\_to\_date\_count (nipyapi.nifi.models.controller\_status\_dto.ControllerStatusDTO(nipyapi.registry.apis.counters\_api.CountersApi  
 attribute), 276  
 up\_to\_date\_count (nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO  
 attribute), 354  
 up\_to\_date\_count (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity  
 attribute), 360  
 update\_access\_policy() (in module  
 nipyapi.security), 22  
 update\_access\_policy()  
 (nipyapi.nifi.apis.policies\_api.PoliciesApi  
 method), 115  
 update\_access\_policy()  
 (nipyapi.registry.apis.policies\_api.PoliciesApi  
 method), 562  
 update\_access\_policy\_with\_http\_info()  
 (nipyapi.nifi.apis.policies\_api.PoliciesApi  
 method), 115  
 update\_access\_policy\_with\_http\_info()  
 method), 562  
 update\_bucket\_with\_http\_info()  
 (nipyapi.registry.apis.buckets\_api.BucketsApi  
 method), 553  
 update\_connection()  
 (nipyapi.nifi.apis.connections\_api.ConnectionsApi  
 method), 42  
 update\_controller\_service\_with\_http\_info()  
 (nipyapi.nifi.apis.connections\_api.ConnectionsApi  
 method), 42  
 update\_controller() (in module nipyapi.canvas),  
 17  
 update\_controller\_config()  
 (nipyapi.nifi.apis.controller\_api.ControllerApi  
 method), 49  
 update\_controller\_config\_with\_http\_info()  
 (nipyapi.nifi.apis.controller\_api.ControllerApi  
 method), 49  
 update\_controller\_service()  
 (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi  
 method), 58  
 update\_controller\_service\_references()  
 (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi  
 method), 58  
 update\_controller\_service\_references\_with\_http\_info()  
 (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi  
 method), 58  
 update\_controller\_service\_with\_http\_info()  
 (nipyapi.nifi.apis.controller\_services\_api.ControllerServicesApi  
 method), 58  
 update\_counter() (nipyapi.nifi.apis.counters\_api.CountersApi  
 method), 60  
 update\_counter\_with\_http\_info()  
 (nipyapi.nifi.apis.counters\_api.CountersApi  
 method), 60  
 update\_descendant\_versioned\_flows  
 (nipyapi.nifi.models.versioned\_flow\_snapshot\_entity.VersionedFlowSnapshotEntity  
 attribute), 567  
 update\_flow() (nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi  
 method), 550  
 update\_flow\_ver() (in module nipyapi.versioning),  
 33  
 update\_flow\_version()  
 (nipyapi.nifi.apis.versions\_api.VersionsApi  
 method), 189  
 update\_flow\_version\_with\_http\_info()  
 (nipyapi.nifi.apis.versions\_api.VersionsApi  
 method), 189  
 update\_flow\_with\_http\_info()  
 (nipyapi.registry.apis.bucket\_flows\_api.BucketFlowsApi



`method`), 550  
`update_funnel()` (*nipyapi.nifi.apis.funnel\_api.FunnelApi*  
`method`), 97  
`update_funnel_with_http_info()`  
(*nipyapi.nifi.apis.funnel\_api.FunnelApi*  
`method`), 97  
`update_input_port()`  
(*nipyapi.nifi.apis.input\_ports\_api.InputPortsApi*  
`method`), 99  
`update_input_port_with_http_info()`  
(*nipyapi.nifi.apis.input\_ports\_api.InputPortsApi*  
`method`), 99  
`update_label()` (*nipyapi.nifi.apis.labels\_api.LabelsApi*  
`method`), 102  
`update_label_with_http_info()`  
(*nipyapi.nifi.apis.labels\_api.LabelsApi*  
`method`), 102  
`update_node()` (*nipyapi.nifi.apis.controller\_api.ControllerApi*  
`method`), 50  
`update_node_with_http_info()`  
(*nipyapi.nifi.apis.controller\_api.ControllerApi*  
`method`), 50  
`update_output_port()`  
(*nipyapi.nifi.apis.output\_ports\_api.OutputPortsApi*  
`method`), 104  
`update_output_port_with_http_info()`  
(*nipyapi.nifi.apis.output\_ports\_api.OutputPortsApi*  
`method`), 104  
`update_parameter_context()` (in module  
*nipyapi.parameters*), 20  
`update_parameter_context()`  
(*nipyapi.nifi.apis.parameter\_contexts\_api.ParameterContextsApi*  
`method`), 112  
`update_parameter_context_with_http_info()`  
(*nipyapi.nifi.apis.parameter\_contexts\_api.ParameterContextsApi*  
`method`), 112  
`update_params_for_auth()`  
(*nipyapi.nifi.api\_client.ApiClient*  
`method`), 535  
`update_params_for_auth()`  
(*nipyapi.registry.api\_client.ApiClient*  
`method`), 621  
`update_process_group()` (in module  
*nipyapi.canvas*), 18  
`update_process_group()`  
(*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
`method`), 137  
`update_process_group_with_http_info()`  
(*nipyapi.nifi.apis.process\_groups\_api.ProcessGroupsApi*  
`method`), 137  
`update_processor()` (in module *nipyapi.canvas*),  
13  
`update_processor()`  
(*nipyapi.nifi.apis.processors\_api.ProcessorsApi*  
`method`), 147  
`update_processor_with_http_info()`  
(*nipyapi.nifi.apis.processors\_api.ProcessorsApi*  
`method`), 147  
`update_registry_client()`  
(*nipyapi.nifi.apis.controller\_api.ControllerApi*  
`method`), 50  
`update_registry_client_with_http_info()`  
(*nipyapi.nifi.apis.controller\_api.ControllerApi*  
`method`), 51  
`update_remote_process_group()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 157  
`update_remote_process_group_input_port()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 157  
`update_remote_process_group_input_port_run_status()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 158  
`update_remote_process_group_input_port_run_status_v1()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 158  
`update_remote_process_group_input_port_with_http_info()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 158  
`update_remote_process_group_output_port()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 159  
`update_remote_process_group_output_port_run_status()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 159  
`update_remote_process_group_output_port_run_status_v1()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 159  
`update_remote_process_group_output_port_with_http_info()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 160  
`update_remote_process_group_run_status()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 160  
`update_remote_process_group_run_status_with_http_info()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 160  
`update_remote_process_group_run_statuses()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 161  
`update_remote_process_group_run_statuses_with_http_info()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 161  
`update_remote_process_group_with_http_info()`  
(*nipyapi.nifi.apis.remote\_process\_groups\_api.RemoteProcessGroupsApi*  
`method`), 161  
`update_reporting_task()`  
(*nipyapi.nifi.apis.reporting\_tasks\_api.ReportingTasksApi*  
`method`), 161

`method`), 168  
`update_reporting_task_with_http_info()`  
     (`nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`  
     `method`), 168  
`update_run_status()`  
     (`nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`  
     `method`), 59  
`update_run_status()`  
     (`nipyapi.nifi.apis.input_ports_api.InputPortsApi`  
     `method`), 100  
`update_run_status()`  
     (`nipyapi.nifi.apis.output_ports_api.OutputPortsApi`  
     `method`), 104  
`update_run_status()`  
     (`nipyapi.nifi.apis.processors_api.ProcessorsApi`  
     `method`), 148  
`update_run_status()`  
     (`nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`  
     `method`), 168  
`update_run_status_with_http_info()`  
     (`nipyapi.nifi.apis.controller_services_api.ControllerServicesApi`  
     `method`), 59  
`update_run_status_with_http_info()`  
     (`nipyapi.nifi.apis.input_ports_api.InputPortsApi`  
     `method`), 100  
`update_run_status_with_http_info()`  
     (`nipyapi.nifi.apis.output_ports_api.OutputPortsApi`  
     `method`), 105  
`update_run_status_with_http_info()`  
     (`nipyapi.nifi.apis.processors_api.ProcessorsApi`  
     `method`), 148  
`update_run_status_with_http_info()`  
     (`nipyapi.nifi.apis.reporting_tasks_api.ReportingTasksApi`  
     `method`), 168  
`update_snippet()` (`nipyapi.nifi.apis.snippets_api.SnippetsApi`  
     `method`), 172  
`update_snippet_with_http_info()`  
     (`nipyapi.nifi.apis.snippets_api.SnippetsApi`  
     `method`), 172  
`update_steps` (`nipyapi.nifi.models.variable_registry_update_request_variable_registry_update_request_dto`  
     `attribute`), 485  
`update_user()` (`nipyapi.nifi.apis.tenants_api.TenantsApi`  
     `method`), 180  
`update_user()` (`nipyapi.registry.apis.tenants_api.TenantsApi`  
     `method`), 567  
`update_user_group()`  
     (`nipyapi.nifi.apis.tenants_api.TenantsApi`  
     `method`), 180  
`update_user_group()`  
     (`nipyapi.registry.apis.tenants_api.TenantsApi`  
     `method`), 567  
`update_user_group_with_http_info()`  
     (`nipyapi.nifi.apis.tenants_api.TenantsApi`  
     `method`), 180  
`update_user_group_with_http_info()`  
     (`nipyapi.registry.apis.tenants_api.TenantsApi`  
     `method`), 567  
`update_user_with_http_info()`  
     (`nipyapi.nifi.apis.tenants_api.TenantsApi`  
     `method`), 180  
`update_user_with_http_info()`  
     (`nipyapi.registry.apis.tenants_api.TenantsApi`  
     `method`), 568  
`update_variable_registry()` (in module  
     `nipyapi.canvas`), 14  
`update_variable_registry()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 138  
`update_variable_registry_with_http_info()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 138  
`update_version_control_request()`  
     (`nipyapi.nifi.apis.versions_api.VersionsApi`  
     `method`), 190  
`update_version_control_request_with_http_info()`  
     (`nipyapi.nifi.apis.versions_api.VersionsApi`  
     `method`), 190  
`UpdateControllerServiceReferenceRequestEntity`  
     (class in `nipyapi.nifi.models.update_controller_service_reference_entity`), 472  
`upload_process_group()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 138  
`upload_process_group_with_http_info()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 139  
`use_template()` (in module `nipyapi.templates`), 26  
`use_template()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 139  
`upload_template_with_http_info()`  
     (`nipyapi.nifi.apis.process_groups_api.ProcessGroupsApi`  
     `method`), 139  
`use_variable_registry_update_request_dto`  
     (class in `nipyapi.nifi.models.variable_registry_update_request_dto`), 485  
`upsert_parameter_to_context()` (in module  
     `nipyapi.parameters`), 21  
`uptime` (`nipyapi.nifi.models.system_diagnostics_snapshot_dto.SystemDiagnosticsSnapshotDto`  
     `attribute`), 464  
`uri` (`nipyapi.nifi.models.about_dto.AboutDTO` `attribute`), 192  
`uri` (`nipyapi.nifi.models.access_policy_entity.AccessPolicyEntity`  
     `attribute`), 197  
`uri` (`nipyapi.nifi.models.access_policy_summary_entity.AccessPolicySummaryEntity`  
     `attribute`), 200  
`uri` (`nipyapi.nifi.models.affected_component_entity.AffectedComponentEntity`  
     `attribute`), 209  
`uri` (`nipyapi.nifi.models.component_reference_entity.ComponentReferenceEntity`  
     `attribute`), 233

uri (nipyapi.nifi.models.connection\_entity.ConnectionEntity attribute), 244

uri (nipyapi.nifi.models.controller\_service\_entity.ControllerServiceEntity attribute), 266

uri (nipyapi.nifi.models.controller\_service\_referencing\_component\_entity.ControllerServiceReferencingComponentEntity attribute), 271

uri (nipyapi.nifi.models.drop\_request\_dto.DropRequestDTO attribute), 291

uri (nipyapi.nifi.models.flow\_file\_dto.FlowFileDTO attribute), 301

uri (nipyapi.nifi.models.flow\_file\_summary\_dto.FlowFileSummaryDTO attribute), 303

uri (nipyapi.nifi.models.funnel\_entity.FunnelEntity attribute), 307

uri (nipyapi.nifi.models.label\_entity.LabelEntity attribute), 315

uri (nipyapi.nifi.models.lineage\_dto.LineageDTO attribute), 317

uri (nipyapi.nifi.models.listing\_request\_dto.ListingRequestDTO attribute), 322

uri (nipyapi.nifi.models.port\_entity.PortEntity attribute), 341

uri (nipyapi.nifi.models.process\_group\_entity.ProcessGroupEntity attribute), 360

uri (nipyapi.nifi.models.process\_group\_flow\_dto.ProcessGroupFlowDTO attribute), 361

uri (nipyapi.nifi.models.processor\_entity.ProcessorEntity attribute), 381

uri (nipyapi.nifi.models.provenance\_dto.ProvenanceDTO attribute), 394

uri (nipyapi.nifi.models.registry\_client\_entity.RegistryClientEntity attribute), 413

uri (nipyapi.nifi.models.registry\_dto.RegistryDTO attribute), 414

uri (nipyapi.nifi.models.remote\_process\_group\_entity.RemoteProcessGroupEntity attribute), 423

uri (nipyapi.nifi.models.remote\_process\_group\_port\_entity.RemoteProcessGroupPortEntity attribute), 428

uri (nipyapi.nifi.models.reporting\_task\_entity.ReportingTaskEntity attribute), 440

uri (nipyapi.nifi.models.snippet\_dto.SnippetDTO attribute), 449

uri (nipyapi.nifi.models.template\_dto.TemplateDTO attribute), 466

uri (nipyapi.nifi.models.template\_entity.TemplateEntity attribute), 467

uri (nipyapi.nifi.models.tenant\_entity.TenantEntity attribute), 471

uri (nipyapi.nifi.models.user\_entity.UserEntity attribute), 476

uri (nipyapi.nifi.models.user\_group\_entity.UserGroupEntity attribute), 479

uri (nipyapi.nifi.models.variable\_registry\_update\_request\_dto.VariableRegistryUpdateRequestDTO attribute), 486

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO attribute), 510

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.Restriction (nipyapi.nifi.models.documented\_type\_dto.DocumentedTypeDTO attribute), 528

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity use\_compression (nipyapi.nifi.models.remote\_process\_group\_port\_dto attribute), 426

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity use\_compression (nipyapi.nifi.models.versioned\_remote\_group\_port\_v1 attribute), 529

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity use\_compression (nipyapi.registry.models.versioned\_remote\_group\_port\_v1 attribute), 615

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_heap (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO attribute), 464

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_heap\_bytes (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO attribute), 464

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_non\_heap (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO attribute), 464

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_non\_heap\_bytes (nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO attribute), 465

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_space (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO attribute), 457

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity used\_space\_bytes (nipyapi.nifi.models.storage\_usage\_dto.StorageUsageDTO attribute), 457

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user (nipyapi.registry.models.user), 583

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_access\_control (nipyapi.nifi.models.port\_dto.PortDTO attribute), 339

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_agent (nipyapi.nifi.api\_client.ApiClient attribute), 535

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_agent (nipyapi.registry.api\_client.ApiClient attribute), 621

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO attribute), 195

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.nifi.models.tenants\_entity.TenantsEntity attribute), 471

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.nifi.models.user\_dto.UserDTO attribute), 475

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.nifi.models.user\_groups\_entity.UserGroupsEntity attribute), 480

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.registry.models.access\_policy.AccessPolicy attribute), 569

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_groups (nipyapi.registry.models.user.User attribute), 584

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_identity (nipyapi.nifi.models.action\_dto.ActionDTO attribute), 203

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity user\_identity (nipyapi.nifi.models.previous\_value\_dto.PreviousValueDTO attribute), 348

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity UserDTO (class in nipyapi.nifi.models.user\_dto), 474

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity UserEntity (class in nipyapi.nifi.models.user\_entity), 475

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity UserGroup (class in nipyapi.nifi.models.user\_group\_entity), 479

uri (nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.RemoteProcessGroupPortEntity UserGroupDTO (class in nipyapi.registry.models.user\_group\_dto), 584

[nipyapi.nifi.models.user\\_group\\_dto](#)), 477  
 UserGroupEntity (class in [nipyapi.nifi.models.controller\\_service\\_dto.ControllerServiceDTO](#) (attribute), 264  
[nipyapi.nifi.models.user\\_group\\_entity](#)), 478 validation\_status  
 UserGroupsEntity (class in [nipyapi.nifi.models.processor\\_dto.ProcessorDTO](#) (attribute), 379  
[nipyapi.nifi.models.user\\_groups\\_entity](#)), validation\_status  
 479  
 users ([nipyapi.nifi.models.access\\_policy\\_dto.AccessPolicyDTO](#) (attribute), 195  
[nipyapi.nifi.models.remote\\_process\\_group\\_status\\_dto.RemoteProcessGroupStatusDTO](#) (attribute), 429  
[nipyapi.nifi.models.tenants\\_entity.TenantsEntity](#) validation\_status  
 attribute), 471 ([nipyapi.nifi.models.reporting\\_task\\_dto.ReportingTaskDTO](#) (attribute), 438  
 users ([nipyapi.nifi.models.user\\_group\\_dto.UserGroupDTO](#) value ([nipyapi.nifi.models.allowable\\_value\\_dto.AllowableValueDTO](#) (attribute), 210  
 attribute), 478 ([nipyapi.nifi.models.attribute\\_dto.AttributeDTO](#) (attribute), 211  
 users ([nipyapi.registry.models.access\\_policy.AccessPolicy](#) value ([nipyapi.nifi.models.counter\\_dto.CounterDTO](#) (attribute), 278  
 attribute), 569 ([nipyapi.nifi.models.state\\_entry\\_dto.StateEntryDTO](#) (attribute), 451  
 users ([nipyapi.registry.models.user\\_group.UserGroup](#) value ([nipyapi.nifi.models.variable\\_dto.VariableDTO](#) (attribute), 481  
 attribute), 585 ([nipyapi.nifi.models.counter\\_dto.CounterDTO](#) value\_count ([nipyapi.nifi.models.counter\\_dto.CounterDTO](#) (attribute), 278  
 UsersEntity (class in [nipyapi.nifi.models.variable\\_entity.VariableEntity](#) (attribute), 482  
[nipyapi.nifi.models.users\\_entity](#)), 480 ([nipyapi.nifi.models.variable\\_registry](#) variable\_registry  
 utilization ([nipyapi.nifi.models.storage\\_usage\\_dto.StorageUsageDTO](#) ([nipyapi.nifi.models.variable\\_registry\\_entity.VariableRegistryEntity](#) (attribute), 484  
 attribute), 457  
 uuid ([nipyapi.nifi.models.flow\\_file\\_dto.FlowFileDTO](#) VariableDTO (class in  
 attribute), 301 ([nipyapi.nifi.models.variable\\_dto](#)), 481  
 uuid ([nipyapi.nifi.models.flow\\_file\\_summary\\_dto.FlowFileSummaryDTO](#) VariableEntity (class in  
 attribute), 303 ([nipyapi.nifi.models.variable\\_entity](#)), 482  
 uuid ([nipyapi.nifi.models.lineage\\_request\\_dto.LineageRequestDTO](#) VariableRegistryDTO (class in  
 attribute), 319 ([nipyapi.nifi.models.variable\\_registry\\_dto](#)), 482  
 VariableRegistryEntity (class in  
 validate\_parameters\_versioning\_support () VariableRegistryUpdateRequestDTO (class in  
 (in module [nipyapi.utils](#)), 31 ([nipyapi.nifi.models.variable\\_registry\\_update\\_request\\_dto](#)),  
 validation\_errors VariableRegistryUpdateRequestEntity (class in  
 ([nipyapi.nifi.models.affected\\_component\\_dto.AffectedComponentDTO](#) ([nipyapi.nifi.models.variable\\_registry\\_update\\_request\\_entity](#)),  
 attribute), 207 486  
 validation\_errors VariableRegistryUpdateStepDTO (class in  
 ([nipyapi.nifi.models.controller\\_service\\_dto.ControllerServiceDTO](#) ([nipyapi.nifi.models.variable\\_registry\\_update\\_step\\_dto](#)),  
 attribute), 264 487  
 validation\_errors variables ([nipyapi.nifi.models.process\\_group\\_dto.ProcessGroupDTO](#) (attribute), 370  
 ([nipyapi.nifi.models.controller\\_service\\_referencing\\_component\\_dto.ControllerServiceReferencingComponentDTO](#)), 484  
 attribute), 269  
 validation\_errors VariableRegistryUpdateStepEntity (class in  
 ([nipyapi.nifi.models.port\\_dto.PortDTO](#) attribute), 339 ([nipyapi.nifi.models.variable\\_registry\\_update\\_step\\_entity](#)), 486  
 validation\_errors VariableRegistryUpdateStepDTO (class in  
 ([nipyapi.nifi.models.processor\\_dto.ProcessorDTO](#) ([nipyapi.nifi.models.variable\\_registry\\_update\\_step\\_dto](#)),  
 attribute), 379 487  
 validation\_errors variables ([nipyapi.nifi.models.process\\_group\\_dto.ProcessGroupDTO](#) (attribute), 370  
 ([nipyapi.nifi.models.remote\\_process\\_group\\_dto.RemoteProcessGroupDTO](#) ([nipyapi.nifi.models.variable\\_registry\\_dto.VariableRegistryDTO](#) (attribute), 483  
 attribute), 421 ([nipyapi.nifi.models.versioned\\_process\\_group.VersionedProcessGroup](#) (attribute), 521  
 validation\_errors ([nipyapi.nifi.models.reporting\\_task\\_dto.ReportingTaskDTO](#) variables ([nipyapi.registry.models.versioned\\_process\\_group.VersionedProcessGroup](#) (attribute), 521  
 attribute), 438  
 validation\_status variables ([nipyapi.registry.models.versioned\\_process\\_group.VersionedProcessGroup](#) (attribute), 521



attribute), 607  
 version (*nipyapi.nifi.models.about\_dto.AboutDTO* attribute), 192  
 version (*nipyapi.nifi.models.bundle.Bundle* attribute), 222  
 version (*nipyapi.nifi.models.bundle\_dto.BundleDTO* attribute), 223  
 version (*nipyapi.nifi.models.revision\_dto.RevisionDTO* attribute), 443  
 version (*nipyapi.nifi.models.version\_control\_information\_dto.VersionControlInformationDTO* attribute), 490  
 version (*nipyapi.nifi.models.versioned\_flow\_coordinates.VersionedFlowCoordinates* attribute), 502  
 version (*nipyapi.registry.models.bundle.Bundle* attribute), 575  
 version (*nipyapi.registry.models.versioned\_flow\_coordinates.VersionedFlowCoordinates* attribute), 595  
 version (*nipyapi.registry.models.versioned\_flow\_snapshot\_metadata.VersionedFlowSnapshotMetadata* attribute), 598  
 version\_control\_component\_mapping (*nipyapi.nifi.models.version\_control\_component\_mapping\_entity.VersionControlComponentMappingEntity* attribute), 488  
 version\_control\_information (*nipyapi.nifi.models.flow\_breadcrumb\_dto.FlowBreadcrumbDTO* attribute), 292  
 version\_control\_information (*nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO* attribute), 354  
 version\_control\_information (*nipyapi.nifi.models.version\_control\_component\_mapping\_entity.VersionControlComponentMappingEntity* attribute), 488  
 version\_control\_information (*nipyapi.nifi.models.version\_control\_information\_entity.VersionControlInformationEntity* attribute), 491  
 version\_control\_information (*nipyapi.nifi.models.versioned\_flow\_update\_request\_dto.VersionedFlowUpdateRequestDTO* attribute), 510  
 version\_count (*nipyapi.nifi.models.versioned\_flow.VersionedFlow* attribute), 501  
 version\_count (*nipyapi.registry.models.versioned\_flow.VersionedFlow* attribute), 594  
 version\_info (*nipyapi.nifi.models.system\_diagnostics\_snapshot\_dto.SystemDiagnosticsSnapshotDTO* attribute), 465  
 VersionControlComponentMappingEntity (class in *nipyapi.nifi.models.version\_control\_component\_mapping\_entity*), 488  
 VersionControlInformationDTO (class in *nipyapi.nifi.models.version\_control\_information\_dto*), 489  
 VersionControlInformationEntity (class in *nipyapi.nifi.models.version\_control\_information\_entity*), 491  
 versioned\_component\_id (*nipyapi.nifi.models.access\_policy\_summary\_dto.AccessPolicySummaryDTO* attribute), 198  
 versioned\_component\_id (*nipyapi.nifi.models.component\_reference\_dto.ComponentReferenceDTO* attribute), 231  
 versioned\_component\_id (*nipyapi.nifi.models.connectable\_dto.ConnectableDTO* attribute), 238  
 versioned\_component\_id (*nipyapi.nifi.models.connection\_dto.ConnectionDTO* attribute), 241  
 versioned\_component\_id (*nipyapi.nifi.models.controller\_service\_dto.ControllerServiceDTO* attribute), 264  
 versioned\_component\_id (*nipyapi.nifi.models.flow\_snapshot\_metadata\_dto.FlowSnapshotMetadataDTO* attribute), 306  
 versioned\_component\_id (*nipyapi.nifi.models.port\_dto.PortDTO* attribute), 340  
 versioned\_component\_id (*nipyapi.nifi.models.process\_group\_dto.ProcessGroupDTO* attribute), 354  
 versioned\_component\_id (*nipyapi.nifi.models.reporting\_task\_dto.ReportingTaskDTO* attribute), 438  
 versioned\_component\_id (*nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity* attribute), 450  
 versioned\_component\_id (*nipyapi.nifi.models.user\_dto.UserDTO* attribute), 475  
 versioned\_component\_id (*nipyapi.nifi.models.user\_group\_dto.UserGroupDTO* attribute), 478  
 versioned\_flow (*nipyapi.nifi.models.start\_version\_control\_request\_entity.StartVersionControlRequestEntity* attribute), 450  
 versioned\_flow (*nipyapi.nifi.models.versioned\_flow\_entity.VersionedFlowEntity* attribute), 504  
 versioned\_flow\_coordinates (*nipyapi.nifi.models.access\_policy\_dto.AccessPolicyDTO* attribute), 195

<code>(nipyapi.nifi.models.versioned_process_group.VersionedProcessGroupEntity.attribute), 521</code>	<code>nipyapi.nifi.models.versioned_flow_dto), 503</code>		
<code>versioned_flow_coordinates</code>	<code>VersionedFlowEntity</code>	<code>(class in nipyapi.registry.models.versioned_process_group.VersionedProcessGroupEntity.attribute), 607</code>	<code>(class in nipyapi.nifi.models.versioned_flow_entity), 504</code>
<code>versioned_flow_snapshot</code>	<code>VersionedFlowsEntity</code>	<code>(class in nipyapi.nifi.models.process_group_entity.ProcessGroupEntity.attribute), 360</code>	<code>(class in nipyapi.nifi.models.versioned_flows_entity), 511</code>
<code>versioned_flow_snapshot</code>	<code>VersionedFlowSnapshot</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity.attribute), 507</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_entity.VersionedFlowSnapshotEntity.attribute), 505</code>
<code>versioned_flow_snapshot_metadata</code>	<code>VersionedFlowSnapshot</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity.attribute), 508</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity.attribute), 595</code>
<code>versioned_flow_snapshot_metadata_set</code>	<code>VersionedFlowSnapshotEntity</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_set_entity.VersionedFlowSnapshotMetadataSetEntity.attribute), 508</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_set_entity.VersionedFlowSnapshotMetadataSetEntity.attribute), 506</code>
<code>versioned_flow_state</code>	<code>VersionedFlowSnapshotMetadata</code>	<code>(class in nipyapi.nifi.models.flow_breadcrumb_entity.FlowBreadcrumbEntity.attribute), 293</code>	<code>(class in nipyapi.registry.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity.attribute), 597</code>
<code>versioned_flow_state</code>	<code>VersionedFlowSnapshotMetadataEntity</code>	<code>(class in nipyapi.nifi.models.process_group_entity.ProcessGroupEntity.attribute), 360</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataEntity.attribute), 507</code>
<code>versioned_flow_state</code>	<code>VersionedFlowSnapshotMetadataSetEntity</code>	<code>(class in nipyapi.nifi.models.process_group_status_snapshot_dto.ProcessGroupStatusSnapshotDto.attribute), 370</code>	<code>(class in nipyapi.nifi.models.versioned_flow_snapshot_metadata_entity.VersionedFlowSnapshotMetadataSetEntity.attribute), 508</code>
<code>versioned_flows</code>	<code>VersionedFlowsEntity</code>	<code>(class in nipyapi.nifi.models.versioned_flows_entity.VersionedFlowsEntity.attribute), 511</code>	<code>(class in nipyapi.nifi.models.versioned_flow_update_request_dto), 500</code>
<code>versioned_group</code>	<code>ComponentSearchResultDTO</code>	<code>(class in nipyapi.nifi.models.component_search_result_entity.ComponentSearchResultEntity.attribute), 234</code>	<code>(class in nipyapi.nifi.models.versioned_flow_update_request_dto), 500</code>
<code>VersionedConnection</code>	<code>VersionedFlowUpdateRequestEntity</code>	<code>(class in nipyapi.nifi.models.versioned_flow_update_request_entity), 493</code>	<code>(class in nipyapi.nifi.models.versioned_flow_update_request_entity), 510</code>
<code>VersionedConnection</code>	<code>VersionedFunnel</code>	<code>(class in nipyapi.nifi.models.versioned_connection), 585</code>	<code>(class in nipyapi.nifi.models.versioned_funnel), 512</code>
<code>VersionedControllerService</code>	<code>VersionedFunnel</code>	<code>(class in nipyapi.registry.models.versioned_connection), 585</code>	<code>(class in nipyapi.registry.models.versioned_funnel), 598</code>
<code>VersionedControllerService</code>	<code>VersionedLabel</code>	<code>(class in nipyapi.nifi.models.versioned_controller_service), 497</code>	<code>(class in nipyapi.nifi.models.versioned_label), 513</code>
<code>VersionedControllerService</code>	<code>VersionedLabel</code>	<code>(class in nipyapi.registry.models.versioned_controller_service), 589</code>	<code>(class in nipyapi.registry.models.versioned_label), 600</code>
<code>VersionedFlow</code>	<code>VersionedPort</code>	<code>(class in nipyapi.nifi.models.versioned_flow), 500</code>	<code>(class in nipyapi.nifi.models.versioned_port), 515</code>
<code>VersionedFlow</code>	<code>VersionedPort</code>	<code>(class in nipyapi.registry.models.versioned_flow), 592</code>	<code>(class in nipyapi.registry.models.versioned_port), 602</code>
<code>VersionedFlowCoordinates</code>	<code>VersionedProcessGroup</code>	<code>(class in nipyapi.nifi.models.versioned_flow_coordinates), 502</code>	<code>(class in nipyapi.nifi.models.versioned_process_group), 517</code>
<code>VersionedFlowCoordinates</code>	<code>VersionedProcessGroup</code>	<code>(class in nipyapi.registry.models.versioned_flow_coordinates), 594</code>	<code>(class in nipyapi.registry.models.versioned_process_group), 603</code>
<code>VersionedFlowDTO</code>	<code>VersionedProcessor</code>	<code>(class in nipyapi.nifi.models.versioned_flow_dto), 503</code>	<code>(class in nipyapi.nifi.models.versioned_processor), 518</code>

[nipyapi.nifi.models.versioned\\_processor](#),  
[521](#)
[yield\\_duration\(nipyapi.nifi.models.versioned\\_processor.VersionedProcessor, attribute\)](#), [525](#)  
[VersionedProcessor](#) (class in [nipyapi.nifi.models.versioned\\_remote\\_process\\_group](#),  
[nipyapi.registry.models.versioned\\_processor](#)), [532](#)  
[607](#)
[yield\\_duration\(nipyapi.registry.models.versioned\\_processor.VersionedProcessor, attribute\)](#), [611](#)  
[VersionedPropertyDescriptor](#) (class in [nipyapi.nifi.models.versioned\\_remote\\_process\\_group](#),  
[nipyapi.nifi.models.versioned\\_property\\_descriptor](#)), [618](#)  
[526](#)
[yield\\_duration\(nipyapi.registry.models.versioned\\_remote\\_process\\_group, attribute\)](#), [618](#)  
[VersionedPropertyDescriptor](#) (class in [nipyapi.nifi.models.versioned\\_remote\\_process\\_group](#),  
[nipyapi.registry.models.versioned\\_property\\_descriptor](#)), [612](#)  
[612](#)
[z\\_index\(nipyapi.nifi.models.versioned\\_connection.VersionedConnection, attribute\)](#), [497](#)  
[VersionedRemoteGroupPort](#) (class in [nipyapi.nifi.models.versioned\\_label.VersionedLabel](#),  
[nipyapi.nifi.models.versioned\\_remote\\_group\\_port](#)), [515](#)  
[527](#)
[z\\_index\(nipyapi.nifi.models.versioned\\_label.VersionedLabel, attribute\)](#), [515](#)  
[VersionedRemoteGroupPort](#) (class in [nipyapi.registry.models.versioned\\_connection.VersionedConnection](#),  
[nipyapi.registry.models.versioned\\_remote\\_group\\_port](#)), [589](#)  
[613](#)
[z\\_index\(nipyapi.registry.models.versioned\\_label.VersionedLabel, attribute\)](#), [601](#)  
[VersionedRemoteProcessGroup](#) (class in [nipyapi.nifi.models.versioned\\_remote\\_process\\_group](#)),  
[529](#)  
[VersionedRemoteProcessGroup](#) (class in [nipyapi.registry.models.versioned\\_remote\\_process\\_group](#)),  
[615](#)  
[VersionInfoDTO](#) (class in [nipyapi.nifi.models.version\\_info\\_dto](#)), [492](#)  
[VersionsApi](#) (class in [nipyapi.nifi.apis.versions\\_api](#)),  
[181](#)

## W

[wait\\_to\\_complete\(\)](#) (in module [nipyapi.utils](#)), [29](#)  
[width\(nipyapi.nifi.models.dimensions\\_dto.DimensionsDTO, attribute\)](#), [287](#)  
[width\(nipyapi.nifi.models.label\\_dto.LabelDTO, attribute\)](#), [314](#)  
[width\(nipyapi.nifi.models.versioned\\_label.VersionedLabel, attribute\)](#), [514](#)  
[width\(nipyapi.registry.models.versioned\\_label.VersionedLabel, attribute\)](#), [601](#)  
[written\(nipyapi.nifi.models.process\\_group\\_status\\_snapshot\\_dto.ProcessGroupStatusSnapshotDTO, attribute\)](#), [370](#)  
[written\(nipyapi.nifi.models.processor\\_status\\_snapshot\\_dto.ProcessorStatusSnapshotDTO, attribute\)](#), [387](#)

## X

[x\(nipyapi.nifi.models.position\\_dto.PositionDTO, attribute\)](#), [347](#)

## Y

[y\(nipyapi.nifi.models.position\\_dto.PositionDTO, attribute\)](#), [347](#)  
[yield\\_duration\(nipyapi.nifi.models.processor\\_config\\_dto.ProcessorConfigDTO, attribute\)](#), [375](#)  
[yield\\_duration\(nipyapi.nifi.models.remote\\_process\\_group\\_dto.RemoteProcessGroupDTO, attribute\)](#), [421](#)