

---

# Python Ironic Client Documentation

*Release 4.4.1.dev3*

**OpenStack LLC**

**Mar 25, 2021**



# CONTENTS

<b>1</b>	<b>Contents</b>	<b>3</b>
1.1	ironicclient Python API . . . . .	3
1.2	python-ironicclient User Documentation . . . . .	5
1.3	Creating the Bare Metal service resources from file . . . . .	68
1.4	python-ironicclient Contributor Documentation . . . . .	71
1.5	Full Ironic Client Python API Reference . . . . .	73
<b>2</b>	<b>Indices and tables</b>	<b>149</b>
	<b>Python Module Index</b>	<b>151</b>
	<b>Index</b>	<b>153</b>



This is a client for the OpenStack [Ironic](#) API. It provides:

- a Python API: the `ironicclient` module, and
- command-line interface: `openstack baremetal`



## CONTENTS

### 1.1 ironicclient Python API

The ironicclient python API lets you access ironic, the OpenStack Bare Metal Provisioning Service.

For example, to manipulate nodes, you interact with an `ironicclient.v1.node` object. You obtain access to nodes via attributes of the `ironicclient.v1.client.Client` object.

#### 1.1.1 Usage

##### Get a Client object

First, create an `ironicclient.v1.client.Client` instance by passing your credentials to `ironicclient.client.get_client()`. By default, the Bare Metal Provisioning system is configured so that only administrators (users with admin role) have access.

---

**Note:** Explicit instantiation of `ironicclient.v1.client.Client` may cause errors since it doesn't verify provided arguments, using `ironicclient.client.get_client()` is preferred way to get client object.

---

There are two different sets of credentials that can be used:

```
* ironic endpoint and auth token
* Identity Service (keystone) credentials
```

##### Using ironic endpoint and auth token

An auth token and the ironic endpoint can be used to authenticate:

```
* os_auth_token: authentication token (from Identity Service)
* ironic_url: ironic API endpoint, eg http://ironic.example.org:6385/v1
```

To create the client, you can use the API like so:

```
>>> from ironicclient import client
>>>
>>> kwargs = {'os_auth_token': '3bcc3d3a03f44e3d8377f9247b0ad155',
>>>           'ironic_url': 'http://ironic.example.org:6385/'}
>>> ironic = client.get_client(1, **kwargs)
```

### Using Identity Service (keystone) credentials

These Identity Service credentials can be used to authenticate:

```
* os_username: name of user
* os_password: user's password
* os_auth_url: Identity Service endpoint for authorization
* insecure: Boolean. If True, does not perform X.509 certificate
  validation when establishing SSL connection with identity
  service. default: False (optional)
* os_tenant_{name|id}: name or ID of tenant
```

Also the following parameters are required when using the Identity API v3:

```
* os_user_domain_name: name of a domain the user belongs to,
  usually 'default'
* os_project_domain_name: name of a domain the project belongs to,
  usually 'default'
```

To create a client, you can use the API like so:

```
>>> from ironicclient import client
>>>
>>> kwargs = {'os_username': 'name',
>>>           'os_password': 'password',
>>>           'os_auth_url': 'http://keystone.example.org:5000/',
>>>           'os_project_name': 'project'}
>>> ironic = client.get_client(1, **kwargs)
```

### Perform ironic operations

Once you have an ironic Client, you can perform various tasks:

```
>>> ironic.driver.list() # list of drivers
>>> ironic.node.list() # list of nodes
>>> ironic.node.get(node_uuid) # information about a particular node
```

When the Client needs to propagate an exception, it will usually raise an instance subclassed from `ironicclient.exc.BaseException` or `ironicclient.exc.ClientException`.

Refer to the modules themselves, for more details.

#### 1.1.2 ironicclient Modules

- modindex



## 1.2 python-ironicclient User Documentation

### 1.2.1 baremetal Standalone Command-Line Interface (CLI)

#### Synopsis

```
baremetal [options] <command> [command-options]
```

```
baremetal help <command>
```

#### Description

The standalone `baremetal` tool allows interacting with the Bare Metal service without installing the OpenStack Client tool as in *openstack baremetal Command-Line Interface (CLI)*.

The standalone tool is mostly identical to its OSC counterpart, with two exceptions:

1. No need to prefix commands with `openstack`.
2. No authentication is assumed by default.

Check the *OSC CLI reference* for a list of available commands.

#### Inspector support

The standalone `baremetal` tool optionally supports the low-level bare metal introspection API provided by `ironic-inspector`. If `ironic-inspector-client` is installed, its `commands` are automatically available (also without the `openstack` prefix).

#### Standalone usage

To use the CLI with a standalone bare metal service, you need to provide an endpoint to connect to. It can be done in three ways:

1. Provide an explicit `--os-endpoint` argument, e.g.:

```
$ baremetal --os-endpoint https://ironic.host:6385 node list
```

2. Set the corresponding environment variable, e.g.:

```
$ export OS_ENDPOINT=https://ironic.host:6385
$ baremetal node list
```

3. Populate a `clouds.yaml` file, setting `baremetal_endpoint_override`, e.g.:

```
$ cat ~/.config/openstack/clouds.yaml
clouds:
  ironic:
    auth_type: none
    baremetal_endpoint_override: http://127.0.0.1:6385
$ export OS_CLOUD=ironic
$ baremetal node list
```

*Inspector support* works similarly, but the `clouds.yaml` option is called `baremetal_introspection_endpoint_override`. The two endpoints can be configured simultaneously, e.g.:

```
$ cat ~/.config/openstack/clouds.yaml
clouds:
  ironic:
    auth_type: none
    baremetal_endpoint_override: http://127.0.0.1:6385
    baremetal_introspection_endpoint_override: http://127.0.0.1:5050
$ export OS_CLOUD=ironic
$ baremetal node list
$ baremetal introspection list
```

### Usage with OpenStack

The standalone CLI can also be used with the Bare Metal service installed as part of OpenStack. See *Authentication* for information on the required input.

### 1.2.2 openstack baremetal Command-Line Interface (CLI)

#### Synopsis

```
openstack [options] baremetal <command> [command-options]
```

```
openstack help baremetal <command>
```

#### Description

The OpenStack Client plugin interacts with the Bare Metal service through the `openstack baremetal` command line interface (CLI).

To use the `openstack` CLI, the `OpenStackClient` (`python-openstackclient`) package must be installed. There are two ways to do this:

- along with this `python-ironicclient` package:

```
$ pip install python-ironicclient[cli]
```

- directly:

```
$ pip install python-openstackclient
```

This CLI is provided by `python-openstackclient` and `osc-lib` projects:

- <https://opendev.org/openstack/python-openstackclient>
- <https://opendev.org/openstack/osc-lib>

## Authentication

To use the CLI, you must provide your OpenStack username, password, project, and auth endpoint. You can use configuration options `--os-username`, `--os-password`, `--os-project-id` (or `--os-project-name`), and `--os-auth-url`, or set the corresponding environment variables:

```
$ export OS_USERNAME=user
$ export OS_PASSWORD=password
$ export OS_PROJECT_NAME=project # or OS_PROJECT_ID
$ export OS_PROJECT_DOMAIN_ID=default
$ export OS_USER_DOMAIN_ID=default
$ export OS_IDENTITY_API_VERSION=3
$ export OS_AUTH_URL=http://auth.example.com:5000/identity
```

## Getting help

To get a list of available (sub)commands and options, run:

```
$ openstack help baremetal
```

To get usage and options of a command, run:

```
$ openstack help baremetal <sub-command>
```

## Examples

Get information about the openstack baremetal node create command:

```
$ openstack help baremetal node create
```

Get a list of available drivers:

```
$ openstack baremetal driver list
```

Enroll a node with the ipmi driver:

```
$ openstack baremetal node create --driver ipmi --driver-info ipmi_
↪address=1.2.3.4
```

Get a list of nodes:

```
$ openstack baremetal node list
```

The baremetal API version can be specified via:

- environment variable `OS_BAREMETAL_API_VERSION`:

```
$ export OS_BAREMETAL_API_VERSION=1.25
```

- or optional command line argument `os-baremetal-api-version`:

```
$ openstack baremetal port group list --os-baremetal-api-version 1.25
```

## Command Reference

### Command Reference

List of released CLI commands available in openstack client. These commands can be referenced by doing `openstack help baremetal`.

#### baremetal allocation

##### baremetal allocation create

Create a new baremetal allocation.

```
openstack baremetal allocation create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--resource-class RESOURCE_CLASS]
  [--trait TRAITS]
  [--candidate-node CANDIDATE_NODES]
  [--name NAME]
  [--uuid UUID]
  [--owner OWNER]
  [--extra <key=value>]
  [--wait [<time-out>]]
  [--node NODE]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the `CLIFF_MAX_TERM_WIDTH` environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable `CLIFF_FIT_WIDTH=1` to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--resource-class** <RESOURCE\_CLASS>  
Resource class to request.

- trait** <TRAITS>  
A trait to request. Can be specified multiple times.
- candidate-node** <CANDIDATE\_NODES>  
A candidate node for this allocation. Can be specified multiple times. If at least one is specified, only the provided candidate nodes are considered for the allocation.
- name** <NAME>  
Unique name of the allocation.
- uuid** <UUID>  
UUID of the allocation.
- owner** <OWNER>  
Owner of the allocation.
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- wait** <time-out>  
Wait for the new allocation to become active. An error is returned if allocation fails and wait is used. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.
- node** <NODE>  
Backfill this allocation from the provided node that has already been deployed. Bypasses the normal allocation process.

This command is provided by the python-ironicclient plugin.

### baremetal allocation delete

Unregister baremetal allocation(s).

```
openstack baremetal allocation delete <allocation> [<allocation> ...]
```

#### allocation

Allocations(s) to delete (name or UUID).

This command is provided by the python-ironicclient plugin.

### baremetal allocation list

List baremetal allocations.

```
openstack baremetal allocation list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--limit <limit>]
```

(continues on next page)

(continued from previous page)

```
[--marker <allocation>]
[--sort <key>[:<direction>]]
[--node <node>]
[--resource-class <resource_class>]
[--state <state>]
[--owner <owner>]
[--long | --fields <field> [<field> ...]]
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric
- noindent**  
whether to disable indenting the JSON
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- limit** <limit>  
Maximum number of allocations to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <allocation>  
Allocation UUID (for example, of the last allocation in the list from a previous request). Returns the list of allocations after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified allocation fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.
- node** <node>  
Only list allocations of this node (name or UUID).
- resource-class** <resource\_class>  
Only list allocations with this resource class.
- state** <state>  
Only list allocations in this state.
- owner** <owner>  
Only list allocations with this owner.

**--long**

Show detailed information about the allocations.

**--fields** <field>

One or more allocation fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

**baremetal allocation set**

Set baremetal allocation properties.

```
openstack baremetal allocation set
  [--name <name>]
  [--extra <key=value>]
  <allocation>
```

**--name** <name>

Set the name of the allocation

**--extra** <key=value>

Extra property to set on this allocation (repeat option to set multiple extra properties)

**allocation**

Name or UUID of the allocation

This command is provided by the python-ironicclient plugin.

**baremetal allocation show**

Show baremetal allocation details.

```
openstack baremetal allocation show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--fields <field> [<field> ...]]
  <id>
```

**-f** <FORMATTER>, **--format** <FORMATTER>

the output format, defaults to table

**-c** COLUMN, **--column** COLUMN

specify the column(s) to include, can be repeated to show multiple columns

**--noindent**

whether to disable indenting the JSON

**--prefix** <PREFIX>

add a prefix to all variable names

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--fields** <field>

One or more allocation fields. Only these fields will be fetched from the server.

**id**

UUID or name of the allocation

This command is provided by the python-ironicclient plugin.

### baremetal allocation unset

Unset baremetal allocation properties.

```
openstack baremetal allocation unset
  [--name]
  [--extra <key>]
  <allocation>
```

**--name**

Unset the name of the allocation

**--extra** <key>

Extra property to unset on this baremetal allocation (repeat option to unset multiple extra property).

**allocation**

Name or UUID of the allocation

This command is provided by the python-ironicclient plugin.

### baremetal chassis

#### baremetal chassis create

Create a new chassis.

```
openstack baremetal chassis create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
```

(continues on next page)



(continued from previous page)

```
[--description <description>]
[--extra <key=value>]
[--uuid <uuid>]
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- description** <description>  
Description for the chassis
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- uuid** <uuid>  
Unique UUID of the chassis

This command is provided by the python-ironicclient plugin.

### baremetal chassis delete

Delete a chassis.

```
openstack baremetal chassis delete <chassis> [<chassis> ...]
```

#### chassis

UUIDs of chassis to delete

This command is provided by the python-ironicclient plugin.

## baremetal chassis list

List the chassis.

```
openstack baremetal chassis list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--fields <field> [<field> ...]]
  [--limit <limit>]
  [--long]
  [--marker <chassis>]
  [--sort <key>[:<direction>]]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**--fields** <field>  
One or more chassis fields. Only these fields will be fetched from the server. Cannot be used when long is specified.

**--limit** <limit>  
Maximum number of chassis to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.

**--long**  
Show detailed information about the chassis

- marker** <chassis>  
Chassis UUID (for example, of the last chassis in the list from a previous request). Returns the list of chassis after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified chassis fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.

This command is provided by the python-ironicclient plugin.

## baremetal chassis set

Set chassis properties.

```
openstack baremetal chassis set
  [--description <description>]
  [--extra <key=value>]
  <chassis>
```

- description** <description>  
Set the description of the chassis
- extra** <key=value>  
Extra to set on this chassis (repeat option to set multiple extras)

**chassis**  
UUID of the chassis

This command is provided by the python-ironicclient plugin.

## baremetal chassis show

Show chassis details.

```
openstack baremetal chassis show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--fields <field> [<field> ...]]
  <chassis>
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--fields** <field>

One or more chassis fields. Only these fields will be fetched from the server.

**chassis**

UUID of the chassis

This command is provided by the python-ironicclient plugin.

## baremetal chassis unset

Unset chassis properties.

```
openstack baremetal chassis unset
  [--description]
  [--extra <key>]
  <chassis>
```

**--description**

Clear the chassis description

**--extra** <key>

Extra to unset on this chassis (repeat option to unset multiple extras)

**chassis**

UUID of the chassis

This command is provided by the python-ironicclient plugin.

## baremetal create

### baremetal create

Create resources from files

```
openstack baremetal create <file> [<file> ...]
```

**file**

File (.yaml or .json) containing descriptions of the resources to create. Can be specified multiple times.

This command is provided by the python-ironicclient plugin.

## baremetal driver

### baremetal driver list

List the enabled drivers.

```

openstack baremetal driver list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--type <type>]
  [--long]

```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**--type** <type>  
Type of driver (classic or dynamic). The default is to list all of them.

**--long**  
Show detailed information about the drivers.

This command is provided by the python-ironicclient plugin.

## baremetal driver passthru call

Call a vendor passthru method for a driver.

```
openstack baremetal driver passthru call
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--arg <key=value>]
  [--http-method <http-method>]
  <driver>
  <method>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--arg** <key=value>  
Argument to pass to the passthru method (repeat option to specify multiple arguments).

**--http-method** <http-method>  
The HTTP method to use in the passthru request. One of DELETE, GET, PATCH, POST, PUT. Defaults to POST.

**driver**  
Name of the driver.

**method**  
Vendor passthru method to be called.

This command is provided by the python-ironicclient plugin.

## baremetal driver passthru list

List available vendor passthru methods for a driver.

```

openstack baremetal driver passthru list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <driver>

```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**driver**  
Name of the driver.

This command is provided by the python-ironicclient plugin.

## baremetal driver property list

List the driver properties.

```
openstack baremetal driver property list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <driver>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**driver**  
Name of the driver.

This command is provided by the python-ironicclient plugin.



## baremetal driver raid property list

List a drivers RAID logical disk properties.

```

openstack baremetal driver raid property list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <driver>

```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**driver**  
Name of the driver.

This command is provided by the python-ironicclient plugin.

## baremetal driver show

Show information about a driver.

```
openstack baremetal driver show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  <driver>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**driver**  
Name of the driver.

This command is provided by the python-ironicclient plugin.

## baremetal node

### baremetal node abort

Set provision state of baremetal node to abort

```
openstack baremetal node abort <node>
```

**node**  
Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node add trait

Add traits to a node.

```
openstack baremetal node add trait <node> <trait> [<trait> ...]
```

### node

Name or UUID of the node

### trait

Trait(s) to add

This command is provided by the python-ironicclient plugin.

## baremetal node adopt

Set provision state of baremetal node to adopt

```
openstack baremetal node adopt [--wait [<time-out>]] <node>
```

### --wait <time-out>

Wait for a node to reach the desired state, active. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

### node

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node bios setting list

List a nodes BIOS settings.

```
openstack baremetal node bios setting list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <node>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN

specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**node**

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

### baremetal node bios setting show

Show a specific BIOS setting for a node.

```
openstack baremetal node bios setting show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  <node>
  <setting
  name>
```

**-f** <FORMATTER>, **--format** <FORMATTER>

the output format, defaults to table

**-c** COLUMN, **--column** COLUMN

specify the column(s) to include, can be repeated to show multiple columns

**--noindent**

whether to disable indenting the JSON

**--prefix** <PREFIX>

add a prefix to all variable names

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**node**

Name or UUID of the node

**setting** name

Setting name to show

This command is provided by the python-ironicclient plugin.

## baremetal node boot device set

Set the boot device for a node

```
openstack baremetal node boot device set [--persistent] <node> <device>
```

**--persistent**

Make changes persistent for all future boots

**node**

Name or UUID of the node

**device**

One of bios, cdrom, disk, pxe, safe, wanboot

This command is provided by the python-ironicclient plugin.

## baremetal node boot device show

Show the boot device information for a node

```
openstack baremetal node boot device show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--supported]
  <node>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**

whether to disable indenting the JSON

**--prefix** <PREFIX>

add a prefix to all variable names

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--supported**

Show the supported boot devices

**node**

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

### baremetal node clean

Set provision state of baremetal node to clean

```
openstack baremetal node clean
  [--wait [<time-out>]]
  --clean-steps <clean-steps>
  <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, manageable. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**--clean-steps** <clean-steps>

The clean steps in JSON format. May be the path to a file containing the clean steps; OR -, with the clean steps being read from standard input; OR a string. The value should be a list of clean-step dictionaries; each dictionary should have keys interface and step, and optional key args.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

### baremetal node console disable

Disable console access for a node

```
openstack baremetal node console disable <node>
```

**node**

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node console enable

Enable console access for a node

```
openstack baremetal node console enable <node>
```

### node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node console show

Show console information for a node

```
openstack baremetal node console show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  <node>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

### node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node create

Register a new node with the baremetal service

```
openstack baremetal node create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--chassis-uuid <chassis>]
--driver <driver>
  [--driver-info <key=value>]
  [--property <key=value>]
  [--extra <key=value>]
  [--uuid <uuid>]
  [--name <name>]
  [--bios-interface <bios_interface>]
  [--boot-interface <boot_interface>]
  [--console-interface <console_interface>]
  [--deploy-interface <deploy_interface>]
  [--inspect-interface <inspect_interface>]
  [--management-interface <management_interface>]
  [--network-data <network data>]
  [--network-interface <network_interface>]
  [--power-interface <power_interface>]
  [--raid-interface <raid_interface>]
  [--rescue-interface <rescue_interface>]
  [--storage-interface <storage_interface>]
  [--vendor-interface <vendor_interface>]
  [--resource-class <resource_class>]
  [--conductor-group <conductor_group>]
  [--automated-clean]
  [--owner <owner>]
  [--lessee <lessee>]
  [--description <description>]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable



- print-empty**  
Print empty table if there is no data to show.
- chassis-uuid** <chassis>  
UUID of the chassis that this node belongs to.
- driver** <driver>  
Driver used to control the node [REQUIRED].
- driver-info** <key=value>  
Key/value pair used by the driver, such as out-of-band management credentials. Can be specified multiple times.
- property** <key=value>  
Key/value pair describing the physical characteristics of the node. This is exported to Nova and used by the scheduler. Can be specified multiple times.
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- uuid** <uuid>  
Unique UUID for the node.
- name** <name>  
Unique name for the node.
- bios-interface** <bios\_interface>  
BIOS interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- boot-interface** <boot\_interface>  
Boot interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- console-interface** <console\_interface>  
Console interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- deploy-interface** <deploy\_interface>  
Deploy interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- inspect-interface** <inspect\_interface>  
Inspect interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- management-interface** <management\_interface>  
Management interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- network-data** <network data>  
JSON string or a file or - for stdin to read static network configuration for the baremetal node associated with this ironic node. Format of this file should comply with Nova network data metadata (network\_data.json). Depending on ironic boot interface capabilities being used, network configuration may or may not be served to the node for offline network configuration.
- network-interface** <network\_interface>  
Network interface used for switching node to cleaning/provisioning networks.

- power-interface** <power\_interface>  
Power interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- raid-interface** <raid\_interface>  
RAID interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- rescue-interface** <rescue\_interface>  
Rescue interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- storage-interface** <storage\_interface>  
Storage interface used by the nodes driver.
- vendor-interface** <vendor\_interface>  
Vendor interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.
- resource-class** <resource\_class>  
Resource class for mapping nodes to Nova flavors
- conductor-group** <conductor\_group>  
Conductor group the node will belong to
- automated-clean**  
Enable automated cleaning for the node
- owner** <owner>  
Owner of the node.
- lessee** <lessee>  
Lessee of the node.
- description** <description>  
Description for the node.

This command is provided by the python-ironicclient plugin.

## baremetal node delete

Unregister baremetal node(s)

```
openstack baremetal node delete <node> [<node> ...]
```

### node

Node(s) to delete (name or UUID)

This command is provided by the python-ironicclient plugin.

## baremetal node deploy

Set provision state of baremetal node to deploy

```
openstack baremetal node deploy
  [--wait [<time-out>]]
  [--config-drive <config-drive>]
  <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, active. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**--config-drive** <config-drive>

A gzipped, base64-encoded configuration drive string OR the path to the configuration drive file OR the path to a directory containing the config drive files OR a JSON object to build config drive from. In case its a directory, a config drive will be generated from it. In case its a JSON object with optional keys *meta\_data*, *user\_data* and *network\_data*, a config drive will be generated on the server side (see the bare metal API reference for more details).

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node inject nmi

Inject NMI to baremetal node

```
openstack baremetal node inject nmi <node>
```

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node inspect

Set provision state of baremetal node to inspect

```
openstack baremetal node inspect [--wait [<time-out>]] <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, manageable. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node list

List baremetal nodes

```
openstack baremetal node list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--limit <limit>]
  [--marker <node>]
  [--sort <key>[:<direction>]]
  [--maintenance | --no-maintenance]
  [--retired | --no-retired]
  [--fault <fault>]
  [--associated | --unassociated]
  [--provision-state <provision state>]
  [--driver <driver>]
  [--resource-class <resource class>]
  [--conductor-group <conductor_group>]
  [--conductor <conductor>]
  [--chassis <chassis UUID>]
  [--owner <owner>]
  [--lessee <lessee>]
  [--description-contains <description_contains>]
  [--long | --fields <field> [<field> ...]]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

- limit** <limit>  
Maximum number of nodes to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <node>  
Node UUID (for example, of the last node in the list from a previous request). Returns the list of nodes after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified node fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.
- maintenance**  
Limit list to nodes in maintenance mode
- no-maintenance**  
Limit list to nodes not in maintenance mode
- retired**  
Limit list to retired nodes.
- no-retired**  
Limit list to not retired nodes.
- fault** <fault>  
List nodes in specified fault.
- associated**  
List only nodes associated with an instance.
- unassociated**  
List only nodes not associated with an instance.
- provision-state** <provision state>  
List nodes in specified provision state.
- driver** <driver>  
Limit list to nodes with driver <driver>
- resource-class** <resource class>  
Limit list to nodes with resource class <resource class>
- conductor-group** <conductor\_group>  
Limit list to nodes with conductor group <conductor group>
- conductor** <conductor>  
Limit list to nodes with conductor <conductor>
- chassis** <chassis UUID>  
Limit list to nodes of this chassis
- owner** <owner>  
Limit list to nodes with owner <owner>
- lessee** <lessee>  
Limit list to nodes with lessee <lessee>
- description-contains** <description\_contains>  
Limit list to nodes with description contains <description\_contains>

**--long**

Show detailed information about the nodes.

**--fields** <field>

One or more node fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

### baremetal node maintenance set

Set baremetal node to maintenance mode

```
openstack baremetal node maintenance set [--reason <reason>] <node>
```

**--reason** <reason>

Reason for setting maintenance mode.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

### baremetal node maintenance unset

Unset baremetal node from maintenance mode

```
openstack baremetal node maintenance unset <node>
```

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

### baremetal node manage

Set provision state of baremetal node to manage

```
openstack baremetal node manage [--wait [<time-out>]] <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, manageable. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node passthru call

Call a vendor passthru method for a node

```

openstack baremetal node passthru call
  [--arg <key=value>]
  [--http-method <http-method>]
  <node>
  <method>

```

**--arg** <key=value>

Argument to pass to the passthru method (repeat option to specify multiple arguments)

**--http-method** <http-method>

The HTTP method to use in the passthru request. One of DELETE, GET, PATCH, POST, PUT. Defaults to POST.

**node**

Name or UUID of the node

**method**

Vendor passthru method to be executed

This command is provided by the python-ironicclient plugin.

## baremetal node passthru list

List vendor passthru methods for a node

```

openstack baremetal node passthru list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <node>

```

**-f** <FORMATTER>, **--format** <FORMATTER>

the output format, defaults to table

**-c** COLUMN, **--column** COLUMN

specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>

when to include quotes, defaults to nonnumeric

**--noindent**

whether to disable indenting the JSON

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN

specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**node**

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node power off

Power off a node

```
openstack baremetal node power off
  [--power-timeout <power-timeout>]
  [--soft]
  <node>
```

**--power-timeout** <power-timeout>

Timeout (in seconds, positive integer) to wait for the target power state before erroring out.

**--soft**

Request graceful power-off.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node power on

Power on a node

```
openstack baremetal node power on
  [--power-timeout <power-timeout>]
  <node>
```

**--power-timeout** <power-timeout>

Timeout (in seconds, positive integer) to wait for the target power state before erroring out.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.



## baremetal node provide

Set provision state of baremetal node to provide

```
openstack baremetal node provide [--wait [<time-out>]] <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, available. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node reboot

Reboot baremetal node

```
openstack baremetal node reboot
  [--soft]
  [--power-timeout <power-timeout>]
  <node>
```

**--soft**

Request Graceful reboot.

**--power-timeout** <power-timeout>

Timeout (in seconds, positive integer) to wait for the target power state before erroring out.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node rebuild

Set provision state of baremetal node to rebuild

```
openstack baremetal node rebuild
  [--wait [<time-out>]]
  [--config-drive <config-drive>]
  <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, active. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**--config-drive** <config-drive>

A gzipped, base64-encoded configuration drive string OR the path to the configuration drive file OR the path to a directory containing the config drive files OR a JSON object to build config drive from. In case its a directory, a config drive will be generated from it. In case its a JSON object with optional keys *meta\_data*, *user\_data* and *network\_data*, a config drive will be generated on the server side (see the bare metal API reference for more details).

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

**baremetal node remove trait**

Remove trait(s) from a node.

```
openstack baremetal node remove trait
  [--all]
  <node>
  [<trait> [<trait> ...]]
```

**--all**

Remove all traits

**node**

Name or UUID of the node

**trait**

Trait(s) to remove

This command is provided by the python-ironicclient plugin.

**baremetal node rescue**

Set provision state of baremetal node to rescue

```
openstack baremetal node rescue
  [--wait [<time-out>]]
  --rescue-password <rescue-password>
  <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, rescue. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**--rescue-password** <rescue-password>

The password that will be used to login to the rescue ramdisk. The value should be a non-empty string.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node set

### Set baremetal properties

```

openstack baremetal node set
  [--instance-uuid <uuid>]
  [--name <name>]
  [--chassis-uuid <chassis UUID>]
  [--driver <driver>]
  [--bios-interface <bios_interface> | --reset-bios-interface]
  [--boot-interface <boot_interface> | --reset-boot-interface]
  [--console-interface <console_interface> | --reset-console-interface]
  [--deploy-interface <deploy_interface> | --reset-deploy-interface]
  [--inspect-interface <inspect_interface> | --reset-inspect-interface]
  [--management-interface <management_interface> | --reset-management-
↪interface]
  [--network-interface <network_interface> | --reset-network-interface]
  [--network-data <network data>]
  [--power-interface <power_interface> | --reset-power-interface]
  [--raid-interface <raid_interface> | --reset-raid-interface]
  [--rescue-interface <rescue_interface> | --reset-rescue-interface]
  [--storage-interface <storage_interface> | --reset-storage-interface]
  [--vendor-interface <vendor_interface> | --reset-vendor-interface]
  [--reset-interfaces]
  [--resource-class <resource_class>]
  [--conductor-group <conductor_group>]
  [--automated-clean]
  [--protected]
  [--protected-reason <protected_reason>]
  [--retired]
  [--retired-reason <retired_reason>]
  [--target-raid-config <target_raid_config>]
  [--property <key=value>]
  [--extra <key=value>]
  [--driver-info <key=value>]
  [--instance-info <key=value>]
  [--owner <owner>]
  [--lessee <lessee>]
  [--description <description>]
  <node>

```

**--instance-uuid** <uuid>  
Set instance UUID of node to <uuid>

**--name** <name>  
Set the name of the node

**--chassis-uuid** <chassis UUID>  
Set the chassis for the node

**--driver** <driver>  
Set the driver for the node

**--bios-interface** <bios\_interface>  
Set the BIOS interface for the node

**--reset-bios-interface**  
Reset the BIOS interface to its hardware type default

**--boot-interface** <boot\_interface>  
Set the boot interface for the node

**--reset-boot-interface**  
Reset the boot interface to its hardware type default

**--console-interface** <console\_interface>  
Set the console interface for the node

**--reset-console-interface**  
Reset the console interface to its hardware type default

**--deploy-interface** <deploy\_interface>  
Set the deploy interface for the node

**--reset-deploy-interface**  
Reset the deploy interface to its hardware type default

**--inspect-interface** <inspect\_interface>  
Set the inspect interface for the node

**--reset-inspect-interface**  
Reset the inspect interface to its hardware type default

**--management-interface** <management\_interface>  
Set the management interface for the node

**--reset-management-interface**  
Reset the management interface to its hardware type default

**--network-interface** <network\_interface>  
Set the network interface for the node

**--reset-network-interface**  
Reset the network interface to its hardware type default

**--network-data** <network data>  
JSON string or a file or - for stdin to read static network configuration for the baremetal node associated with this ironic node. Format of this file should comply with Nova network data meta-data (network\_data.json). Depending on ironic boot interface capabilities being used, network configuration may or may not be served to the node for offline network configuration.

**--power-interface** <power\_interface>  
Set the power interface for the node

**--reset-power-interface**  
Reset the power interface to its hardware type default

**--raid-interface** <raid\_interface>  
Set the RAID interface for the node

**--reset-raid-interface**  
Reset the RAID interface to its hardware type default

**--rescue-interface** <rescue\_interface>  
Set the rescue interface for the node

**--reset-rescue-interface**  
Reset the rescue interface to its hardware type default

- storage-interface** <storage\_interface>  
Set the storage interface for the node
- reset-storage-interface**  
Reset the storage interface to its hardware type default
- vendor-interface** <vendor\_interface>  
Set the vendor interface for the node
- reset-vendor-interface**  
Reset the vendor interface to its hardware type default
- reset-interfaces**  
Reset all interfaces not specified explicitly to their default implementations. Only valid with driver.
- resource-class** <resource\_class>  
Set the resource class for the node
- conductor-group** <conductor\_group>  
Set the conductor group for the node
- automated-clean**  
Enable automated cleaning for the node
- protected**  
Mark the node as protected
- protected-reason** <protected\_reason>  
Set the reason of marking the node as protected
- retired**  
Mark the node as retired
- retired-reason** <retired\_reason>  
Set the reason of marking the node as retired
- target-raid-config** <target\_raid\_config>  
Set the target RAID configuration (JSON) for the node. This can be one of: 1. a file containing JSON data of the RAID configuration; 2. - to read the contents from standard input; or 3. a valid JSON string.
- property** <key=value>  
Property to set on this baremetal node (repeat option to set multiple properties)
- extra** <key=value>  
Extra to set on this baremetal node (repeat option to set multiple extras)
- driver-info** <key=value>  
Driver information to set on this baremetal node (repeat option to set multiple driver infos)
- instance-info** <key=value>  
Instance information to set on this baremetal node (repeat option to set multiple instance infos)
- owner** <owner>  
Set the owner for the node
- lessee** <lessee>  
Set the lessee for the node

**--description** <description>  
Set the description for the node

**node**  
Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node show

Show baremetal node details

```
openstack baremetal node show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--instance]
  [--fields <field> [<field> ...]]
  <node>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--instance**  
<node> is an instance UUID.

**--fields** <field>  
One or more node fields. Only these fields will be fetched from the server.

**node**  
Name or UUID of the node (or instance UUID if instance is specified)

This command is provided by the python-ironicclient plugin.

## baremetal node trait list

List a nodes traits.

```
openstack baremetal node trait list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <node>
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**node**  
Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node undeploy

Set provision state of baremetal node to deleted

```
openstack baremetal node undeploy [--wait [<time-out>]] <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, available. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node unrescue

Set provision state of baremetal node to unrescue

```
openstack baremetal node unrescue [--wait [<time-out>]] <node>
```

**--wait** <time-out>

Wait for a node to reach the desired state, active. Optionally takes a timeout value (in seconds). The default value is 0, meaning it will wait indefinitely.

**node**

Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node unset

Unset baremetal properties

```
openstack baremetal node unset
  [--instance-uuid]
  [--name]
  [--resource-class]
  [--target-raid-config]
  [--property <key>]
  [--extra <key>]
  [--driver-info <key>]
  [--instance-info <key>]
  [--chassis-uuid]
  [--bios-interface]
  [--boot-interface]
  [--console-interface]
  [--deploy-interface]
  [--inspect-interface]
  [--network-data]
  [--management-interface]
  [--network-interface]
  [--power-interface]
  [--raid-interface]
```

(continues on next page)



(continued from previous page)

```

[--rescue-interface]
[--storage-interface]
[--vendor-interface]
[--conductor-group]
[--automated-clean]
[--protected]
[--protected-reason]
[--retired]
[--retired-reason]
[--owner]
[--lessee]
[--description]
<node>

```

**--instance-uuid**

Unset instance UUID on this baremetal node

**--name**

Unset the name of the node

**--resource-class**

Unset the resource class of the node

**--target-raid-config**

Unset the target RAID configuration of the node

**--property** <key>

Property to unset on this baremetal node (repeat option to unset multiple properties)

**--extra** <key>

Extra to unset on this baremetal node (repeat option to unset multiple extras)

**--driver-info** <key>

Driver information to unset on this baremetal node (repeat option to unset multiple driver informations)

**--instance-info** <key>

Instance information to unset on this baremetal node (repeat option to unset multiple instance informations)

**--chassis-uuid**

Unset chassis UUID on this baremetal node

**--bios-interface**

Unset BIOS interface on this baremetal node

**--boot-interface**

Unset boot interface on this baremetal node

**--console-interface**

Unset console interface on this baremetal node

**--deploy-interface**

Unset deploy interface on this baremetal node

**--inspect-interface**

Unset inspect interface on this baremetal node

- network-data**  
Unset network data on this baremetal port.
  - management-interface**  
Unset management interface on this baremetal node
  - network-interface**  
Unset network interface on this baremetal node
  - power-interface**  
Unset power interface on this baremetal node
  - raid-interface**  
Unset RAID interface on this baremetal node
  - rescue-interface**  
Unset rescue interface on this baremetal node
  - storage-interface**  
Unset storage interface on this baremetal node
  - vendor-interface**  
Unset vendor interface on this baremetal node
  - conductor-group**  
Unset conductor group for this baremetal node (the default group will be used)
  - automated-clean**  
Unset automated clean option on this baremetal node (the value from configuration will be used)
  - protected**  
Unset the protected flag on the node
  - protected-reason**  
Unset the protected reason (gets unset automatically when protected is unset)
  - retired**  
Unset the retired flag on the node
  - retired-reason**  
Unset the retired reason (gets unset automatically when retired is unset)
  - owner**  
Unset the owner field of the node
  - lessee**  
Unset the lessee field of the node
  - description**  
Unset the description field of the node
- node**  
Name or UUID of the node.

This command is provided by the python-ironicclient plugin.

## baremetal node validate

Validate a nodes driver interfaces

```

openstack baremetal node validate
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <node>

```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated

**node**  
Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal node vif attach

Attach VIF to a given node

```
openstack baremetal node vif attach
  [--port-uuid <port-uuid>]
  [--vif-info <key=value>]
  <node>
  <vif-id>
```

**--port-uuid** <port-uuid>

UUID of the baremetal port to attach the VIF to.

**--vif-info** <key=value>

Record arbitrary key/value metadata. Can be specified multiple times. The mandatory id parameter cannot be specified as a key.

**node**

Name or UUID of the node

**vif-id**

Name or UUID of the VIF to attach to a node.

This command is provided by the python-ironicclient plugin.

## baremetal node vif detach

Detach VIF from a given node

```
openstack baremetal node vif detach <node> <vif-id>
```

**node**

Name or UUID of the node

**vif-id**

Name or UUID of the VIF to detach from a node.

This command is provided by the python-ironicclient plugin.

## baremetal node vif list

Show attached VIFs for a node

```
openstack baremetal node vif list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  <node>
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric
- noindent**  
whether to disable indenting the JSON
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- node**  
Name or UUID of the node

This command is provided by the python-ironicclient plugin.

## baremetal port

### baremetal port create

Create a new port

```
openstack baremetal port create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  --node <uuid>
  [--uuid <uuid>]
  [--extra <key=value>]
  [--local-link-connection <key=value>]
  [-l <key=value>]
  [--pxe-enabled <boolean>]
  [--port-group <uuid>]
  [--physical-network <physical network>]
  [--is-smartnic]
  <address>
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- node** <uuid>  
UUID of the node that this port belongs to.
- uuid** <uuid>  
UUID of the port.
- extra** <key=value>  
Record arbitrary key/value metadata. Argument can be specified multiple times.
- local-link-connection** <key=value>  
Key/value metadata describing Local link connection information. Valid keys are switch\_info, switch\_id, port\_id and hostname. The keys switch\_id and port\_id are required. In case of a Smart NIC port, the required keys are port\_id and hostname. Argument can be specified multiple times.
- l** <key=value>  
DEPRECATED. Please use local-link-connection instead. Key/value metadata describing Local link connection information. Valid keys are switch\_info, switch\_id, and port\_id. The keys switch\_id and port\_id are required. Can be specified multiple times.
- pxe-enabled** <boolean>  
Indicates whether this Port should be used when PXE booting this Node.
- port-group** <uuid>  
UUID of the port group that this port belongs to.
- physical-network** <physical network>  
Name of the physical network to which this port is connected.
- is-smartnic**  
Indicates whether this Port is a Smart NIC port
- address**  
MAC address for this port.

This command is provided by the python-ironicclient plugin.

## baremetal port delete

Delete port(s).

```
openstack baremetal port delete <port> [<port> ...]
```

### port

UUID(s) of the port(s) to delete.

This command is provided by the python-ironicclient plugin.

## baremetal port group create

Create a new baremetal port group.

```
openstack baremetal port group create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  --node <uuid>
  [--address <mac-address>]
  [--name NAME]
  [--uuid UUID]
  [--extra <key=value>]
  [--mode MODE]
  [--property <key=value>]
  [--support-standalone-ports | --unsupport-standalone-ports]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--node** <uuid>  
UUID of the node that this port group belongs to.

- address** <mac-address>  
MAC address for this port group.
- name** <NAME>  
Name of the port group.
- uuid** <UUID>  
UUID of the port group.
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- mode** <MODE>  
Mode of the port group. For possible values, refer to <https://www.kernel.org/doc/Documentation/networking/bonding.txt>.
- property** <key=value>  
Key/value property related to this port groups configuration. Can be specified multiple times.
- support-standalone-ports**  
Ports that are members of this port group can be used as stand-alone ports. (default)
- unsupport-standalone-ports**  
Ports that are members of this port group cannot be used as stand-alone ports.

This command is provided by the python-ironicclient plugin.

### baremetal port group delete

Unregister baremetal port group(s).

```
openstack baremetal port group delete <port group> [<port group> ...]
```

**port** group  
Port group(s) to delete (name or UUID).

This command is provided by the python-ironicclient plugin.

### baremetal port group list

List baremetal port groups.

```
openstack baremetal port group list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--limit <limit>]
  [--marker <port group>]
  [--sort <key>[:<direction>]]
  [--address <mac-address>]
```

(continues on next page)



(continued from previous page)

```
[--node <node>]
[--long | --fields <field> [<field> ...]]
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric
- noindent**  
whether to disable indenting the JSON
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- limit** <limit>  
Maximum number of port groups to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <port group>  
Port group UUID (for example, of the last port group in the list from a previous request). Returns the list of port groups after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified port group fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.
- address** <mac-address>  
Only show information for the port group with this MAC address.
- node** <node>  
Only list port groups of this node (name or UUID).
- long**  
Show detailed information about the port groups.
- fields** <field>  
One or more port group fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

## baremetal port group set

Set baremetal port group properties.

```
openstack baremetal port group set
  [--node <uuid>]
  [--address <mac-address>]
  [--name <name>]
  [--extra <key=value>]
  [--mode MODE]
  [--property <key=value>]
  [--support-standalone-ports | --unsupport-standalone-ports]
  <port
  group>
```

**--node** <uuid>

Update UUID of the node that this port group belongs to.

**--address** <mac-address>

MAC address for this port group.

**--name** <name>

Name of the port group.

**--extra** <key=value>

Extra to set on this baremetal port group (repeat option to set multiple extras).

**--mode** <MODE>

Mode of the port group. For possible values, refer to <https://www.kernel.org/doc/Documentation/networking/bonding.txt>.

**--property** <key=value>

Key/value property related to this port groups configuration (repeat option to set multiple properties).

**--support-standalone-ports**

Ports that are members of this port group can be used as stand-alone ports.

**--unsupport-standalone-ports**

Ports that are members of this port group cannot be used as stand-alone ports.

**port** group

Name or UUID of the port group.

This command is provided by the python-ironicclient plugin.

## baremetal port group show

Show baremetal port group details.

```
openstack baremetal port group show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
```

(continues on next page)

(continued from previous page)

```

[--fit-width]
[--print-empty]
[--address]
[--fields <field> [<field> ...]]
<id>

```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- address**  
<id> is the MAC address (instead of UUID or name) of the port group.
- fields** <field>  
One or more port group fields. Only these fields will be fetched from the server.
- id**  
UUID or name of the port group (or MAC address if address is specified).

This command is provided by the python-ironicclient plugin.

## baremetal port group unset

Unset baremetal port group properties.

```

openstack baremetal port group unset
  [--name]
  [--address]
  [--extra <key>]
  [--property <key>]
  <port
  group>

```

- name**  
Unset the name of the port group.

**--address**

Unset the address of the port group.

**--extra** <key>

Extra to unset on this baremetal port group (repeat option to unset multiple extras).

**--property** <key>

Property to unset on this baremetal port group (repeat option to unset multiple properties).

**port** group

Name or UUID of the port group.

This command is provided by the python-ironicclient plugin.

## baremetal port list

List baremetal ports.

```
openstack baremetal port list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--address <mac-address>]
  [--node <node>]
  [--port-group <port group>]
  [--limit <limit>]
  [--marker <port>]
  [--sort <key>[:<direction>]]
  [--long | --fields <field> [<field> ...]]
```

**-f** <FORMATTER>, **--format** <FORMATTER>

the output format, defaults to table

**-c** COLUMN, **--column** COLUMN

specify the column(s) to include, can be repeated to show multiple columns

**--quote** <QUOTE\_MODE>

when to include quotes, defaults to nonnumeric

**--noindent**

whether to disable indenting the JSON

**--max-width** <integer>

Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- address** <mac-address>  
Only show information for the port with this MAC address.
- node** <node>  
Only list ports of this node (name or UUID).
- port-group** <port group>  
Only list ports of this port group (name or UUID).
- limit** <limit>  
Maximum number of ports to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <port>  
Port UUID (for example, of the last port in the list from a previous request). Returns the list of ports after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified port fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.
- long**  
Show detailed information about ports.
- fields** <field>  
One or more port fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

## baremetal port set

Set baremetal port properties.

```
openstack baremetal port set
  [--node <uuid>]
  [--address <address>]
  [--extra <key=value>]
  [--port-group <uuid>]
  [--local-link-connection <key=value>]
  [--pxe-enabled | --pxe-disabled]
  [--physical-network <physical network>]
  [--is-smartnic]
  <port>
```

- node** <uuid>  
Set UUID of the node that this port belongs to
- address** <address>  
Set MAC address for this port
- extra** <key=value>  
Extra to set on this baremetal port (repeat option to set multiple extras)

- port-group** <uuid>  
Set UUID of the port group that this port belongs to.
  - local-link-connection** <key=value>  
Key/value metadata describing local link connection information. Valid keys are `switch_info`, `switch_id`, `port_id` and `hostname`. The keys `switch_id` and `port_id` are required. In case of a Smart NIC port, the required keys are `port_id` and `hostname`. Argument can be specified multiple times.
  - pxe-enabled**  
Indicates that this port should be used when PXE booting this node (default)
  - pxe-disabled**  
Indicates that this port should not be used when PXE booting this node
  - physical-network** <physical network>  
Set the name of the physical network to which this port is connected.
  - is-smartnic**  
Set port to be Smart NIC port
- port**  
UUID of the port

This command is provided by the `python-ironicclient` plugin.

## baremetal port show

Show baremetal port details.

```
openstack baremetal port show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--address]
  [--fields <field> [<field> ...]]
  <id>
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the `CLIFF_MAX_TERM_WIDTH` environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--address**

<id> is the MAC address (instead of the UUID) of the port.

**--fields <field>**

One or more port fields. Only these fields will be fetched from the server.

**id**

UUID of the port (or MAC address if address is specified).

This command is provided by the python-ironicclient plugin.

## baremetal port unset

Unset baremetal port properties.

```
openstack baremetal port unset
  [--extra <key>]
  [--port-group]
  [--physical-network]
  [--is-smartnic]
  <port>
```

**--extra <key>**

Extra to unset on this baremetal port (repeat option to unset multiple extras)

**--port-group**

Remove port from the port group

**--physical-network**

Unset the physical network on this baremetal port.

**--is-smartnic**

Set Port as not Smart NIC port

**port**

UUID of the port.

This command is provided by the python-ironicclient plugin.

## baremetal volume

### baremetal volume connector create

Create a new baremetal volume connector.

```
openstack baremetal volume connector create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
```

(continues on next page)

(continued from previous page)

```
[--noindent]
[--prefix PREFIX]
[--max-width <integer>]
[--fit-width]
[--print-empty]
--node <uuid>
--type <type>
--connector-id <connector
id>
[--uuid <uuid>]
[--extra <key=value>]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--node** <uuid>  
UUID of the node that this volume connector belongs to.

**--type** <type>  
Type of the volume connector. Can be iqn, ip, mac, wwnn, wwpn, port, portgroup.

**--connector-id** <connector id>  
ID of the volume connector in the specified type. For example, the iSCSI initiator IQN for the node if the type is iqn.

**--uuid** <uuid>  
UUID of the volume connector.

**--extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.

This command is provided by the python-ironicclient plugin.



## baremetal volume connector delete

Unregister baremetal volume connector(s).

```

openstack baremetal volume connector delete
  <volume
  connector>
  [<volume connector> ...]

```

**volume connector**

UUID(s) of the volume connector(s) to delete.

This command is provided by the python-ironicclient plugin.

## baremetal volume connector list

List baremetal volume connectors.

```

openstack baremetal volume connector list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--node <node>]
  [--limit <limit>]
  [--marker <volume connector>]
  [--sort <key>[:<direction>]]
  [--long | --fields <field> [<field> ...]]

```

**-f <FORMATTER>, --format <FORMATTER>**  
the output format, defaults to table

**-c COLUMN, --column COLUMN**  
specify the column(s) to include, can be repeated to show multiple columns

**--quote <QUOTE\_MODE>**  
when to include quotes, defaults to nonnumeric

**--noindent**  
whether to disable indenting the JSON

**--max-width <integer>**  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- node** <node>  
Only list volume connectors of this node (name or UUID).
- limit** <limit>  
Maximum number of volume connectors to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <volume connector>  
Volume connector UUID (for example, of the last volume connector in the list from a previous request). Returns the list of volume connectors after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified volume connector fields and directions (asc or desc) (default:asc). Multiple fields and directions can be specified, separated by comma.
- long**  
Show detailed information about volume connectors.
- fields** <field>  
One or more volume connector fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

### baremetal volume connector set

Set baremetal volume connector properties.

```
openstack baremetal volume connector set
  [--node <uuid>]
  [--type <type>]
  [--connector-id <connector id>]
  [--extra <key=value>]
  <volume
  connector>
```

- node** <uuid>  
UUID of the node that this volume connector belongs to.
  - type** <type>  
Type of the volume connector. Can be iqn, ip, mac, wwnn, wwpn, port, portgroup.
  - connector-id** <connector id>  
ID of the volume connector in the specified type.
  - extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- volume connector**  
UUID of the volume connector.

This command is provided by the python-ironicclient plugin.

## baremetal volume connector show

Show baremetal volume connector details.

```

openstack baremetal volume connector show
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--fields <field> [<field> ...]]
  <id>

```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--fields** <field>  
One or more volume connector fields. Only these fields will be fetched from the server.

**id**  
UUID of the volume connector.

This command is provided by the python-ironicclient plugin.

## baremetal volume connector unset

Unset baremetal volume connector properties.

```

openstack baremetal volume connector unset
  [--extra <key>]
  <volume
connector>

```

**--extra** <key>  
Extra to unset (repeat option to unset multiple extras)

**volume connector**  
UUID of the volume connector.

This command is provided by the python-ironicclient plugin.

### baremetal volume target create

Create a new baremetal volume target.

```
openstack baremetal volume target create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  --node <uuid>
  --type <volume
type>
  [--property <key=value>]
  --boot-index <boot
index>
  --volume-id <volume
id>
  [--uuid <uuid>]
  [--extra <key=value>]
```

**-f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table

**-c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

**--noindent**  
whether to disable indenting the JSON

**--prefix** <PREFIX>  
add a prefix to all variable names

**--max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**  
Print empty table if there is no data to show.

**--node** <uuid>  
UUID of the node that this volume target belongs to.

**--type** <volume type>  
Type of the volume target, e.g. iscsi, fibre\_channel.

- property** <key=value>  
Key/value property related to the type of this volume target. Can be specified multiple times.
- boot-index** <boot index>  
Boot index of the volume target.
- volume-id** <volume id>  
ID of the volume associated with this target.
- uuid** <uuid>  
UUID of the volume target.
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.

This command is provided by the python-ironicclient plugin.

### baremetal volume target delete

Unregister baremetal volume target(s).

```
openstack baremetal volume target delete
  <volume
  target>
  [<volume target> ...]
```

**volume** target  
UUID(s) of the volume target(s) to delete.

This command is provided by the python-ironicclient plugin.

### baremetal volume target list

List baremetal volume targets.

```
openstack baremetal volume target list
  [-f {csv,json,table,value,yaml}]
  [-c COLUMN]
  [--quote {all,minimal,none,nonnumeric}]
  [--noindent]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--sort-column SORT_COLUMN]
  [--node <node>]
  [--limit <limit>]
  [--marker <volume target>]
  [--sort <key>[:<direction>]]
  [--long | --fields <field> [<field> ...]]
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns

- quote** <QUOTE\_MODE>  
when to include quotes, defaults to nonnumeric
- noindent**  
whether to disable indenting the JSON
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.
- fit-width**  
Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable
- print-empty**  
Print empty table if there is no data to show.
- sort-column** SORT\_COLUMN  
specify the column(s) to sort the data (columns specified first have a priority, non-existing columns are ignored), can be repeated
- node** <node>  
Only list volume targets of this node (name or UUID).
- limit** <limit>  
Maximum number of volume targets to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <volume target>  
Volume target UUID (for example, of the last volume target in the list from a previous request). Returns the list of volume targets after this UUID.
- sort** <key>[:<direction>]  
Sort output by specified volume target fields and directions (asc or desc) (default:asc). Multiple fields and directions can be specified, separated by comma.
- long**  
Show detailed information about volume targets.
- fields** <field>  
One or more volume target fields. Only these fields will be fetched from the server. Can not be used when long is specified.

This command is provided by the python-ironicclient plugin.

## baremetal volume target set

Set baremetal volume target properties.

```
openstack baremetal volume target set set
  [--node <uuid>]
  [--type <volume type>]
  [--property <key=value>]
  [--boot-index <boot index>]
  [--volume-id <volume id>]
  [--extra <key=value>]
```

(continues on next page)

(continued from previous page)

```
<volume
target>
```

- node** <uuid>  
UUID of the node that this volume target belongs to.
- type** <volume type>  
Type of the volume target, e.g. iscsi, fibre\_channel.
- property** <key=value>  
Key/value property related to the type of this volume target. Can be specified multiple times.
- boot-index** <boot index>  
Boot index of the volume target.
- volume-id** <volume id>  
ID of the volume associated with this target.
- extra** <key=value>  
Record arbitrary key/value metadata. Can be specified multiple times.
- volume target**  
UUID of the volume target.

This command is provided by the python-ironicclient plugin.

## baremetal volume target show

Show baremetal volume target details.

```
openstack baremetal volume target show
[-f {json,shell,table,value,yaml}]
[-c COLUMN]
[--noindent]
[--prefix PREFIX]
[--max-width <integer>]
[--fit-width]
[--print-empty]
[--fields <field> [<field> ...]]
<id>
```

- f** <FORMATTER>, **--format** <FORMATTER>  
the output format, defaults to table
- c** COLUMN, **--column** COLUMN  
specify the column(s) to include, can be repeated to show multiple columns
- noindent**  
whether to disable indenting the JSON
- prefix** <PREFIX>  
add a prefix to all variable names
- max-width** <integer>  
Maximum display width, <1 to disable. You can also use the CLIFF\_MAX\_TERM\_WIDTH environment variable, but the parameter takes precedence.

**--fit-width**

Fit the table to the display width. Implied if max-width greater than 0. Set the environment variable CLIFF\_FIT\_WIDTH=1 to always enable

**--print-empty**

Print empty table if there is no data to show.

**--fields** <field>

One or more volume target fields. Only these fields will be fetched from the server.

**id**

UUID of the volume target.

This command is provided by the python-ironicclient plugin.

### baremetal volume target unset

Unset baremetal volume target properties.

```
openstack baremetal volume target unset
  [--extra <key>]
  [--property <key>]
  <volume
  target>
```

**--extra** <key>

Extra to unset (repeat option to unset multiple extras)

**--property** <key>

Property to unset on this baremetal volume target (repeat option to unset multiple properties).

**volume** target

UUID of the volume target.

This command is provided by the python-ironicclient plugin.

## 1.3 Creating the Bare Metal service resources from file

It is possible to create a set of resources using their descriptions in JSON or YAML format. It can be done in one of two ways:

1. Using OpenStackClient bare metal plugin CLIs command `openstack baremetal create`:

```
$ openstack -h baremetal create
usage: openstack baremetal create [-h] <file> [<file> ...]

Create resources from files

positional arguments:
  <file>          File (.yaml or .json) containing descriptions of the
                  resources to create. Can be specified multiple times.
```

2. Programmatically using the Python API:



```
ironicclient.v1.create_resources.create_resources(client, file-
                                                names)
```

Create resources using their JSON or YAML descriptions.

#### Parameters

- **client** an instance of ironic client;
- **filenames** a list of filenames containing JSON or YAML resources definitions.

**Raises** ClientException if any operation during files processing/resource creation fails.

### 1.3.1 File containing Resource Descriptions

The resources to be created can be described either in JSON or YAML. A file ending with `.json` is assumed to contain valid JSON, and a file ending with `.yaml` is assumed to contain valid YAML. Specifying a file with any other extension leads to an error.

The resources that can be created are chassis, nodes, port groups and ports. A chassis can contain nodes (and resources of nodes) definitions nested under "nodes" key. A node can contain port groups definitions nested under "portgroups", and ports definitions under "ports" keys. Ports can be also nested under port groups in "ports" key.

The schema used to validate the supplied data is the following:

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description": "Schema for ironic resources file",
  "type": "object",
  "properties": {
    "chassis": {
      "type": "array",
      "items": {
        "type": "object"
      }
    },
    "nodes": {
      "type": "array",
      "items": {
        "type": "object"
      }
    }
  },
  "additionalProperties": False
}
```

More detailed description of the creation process can be seen in the following sections.

### 1.3.2 Examples

Here is an example of the JSON file that can be passed to the `create` command:

```
{
  "chassis": [
    {
      "description": "chassis 3 in row 23",
      "nodes": [
        {
          "name": "node-3",
          "driver": "agent_ipmitool",
          "portgroups": [
            {
              "name": "switch.cz7882.ports.1-2",
              "ports": [
                {
                  "address": "ff:00:00:00:00:00"
                },
                {
                  "address": "ff:00:00:00:00:01"
                }
              ]
            }
          ],
          "ports": [
            {
              "address": "00:00:00:00:00:02"
            },
            {
              "address": "00:00:00:00:00:03"
            }
          ]
        },
        {
          "name": "node-4",
          "driver": "agent_ipmitool",
          "ports": [
            {
              "address": "00:00:00:00:00:04"
            },
            {
              "address": "00:00:00:00:00:01"
            }
          ]
        }
      ]
    }
  ],
  "nodes": [
    {
      "name": "node-5",
      "driver": "pxe_ipmitool",
      "chassis_uuid": "74d93e6e-7384-4994-a614-fd7b399b0785",
      "ports": [
        {
          "address": "00:00:00:00:00:00"
        }
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
    ],
  },
  {
    "name": "node-6",
    "driver": "pxe_ipmitool"
  }
]
```

### 1.3.3 Creation Process

1. The client deserializes the files contents and validates that the top-level dictionary in each of them contains only chassis and/or nodes keys, and their values are lists. The creation process is aborted if any failure is encountered in this stage. The rest of the validation is done by the ironic-api service.
2. Each resource is created via issuing a POST request (with the resources dictionary representation in the body) to the ironic-api service. In the case of nested resources ("nodes" key inside chassis, "portgroups" key inside nodes, "ports" key inside nodes or portgroups), the top-level resource is created first, followed by the sub-resources. For example, if a chassis contains a list of nodes, the chassis will be created first followed by the creation of each node. The same is true for ports and port groups described within nodes.
3. If a resource could not be created, it does not stop the entire process. Any sub-resources of the failed resource will not be created, but otherwise, the rest of the resources will be created if possible. Any failed resources will be mentioned in the response.

## 1.4 python-ironicclient Contributor Documentation

### 1.4.1 Contributing to python-ironicclient

If you're interested in contributing to the python-ironicclient project, the following will help get you started.

#### #openstack-ironic on Freenode IRC Network

There is a very active chat channel at <irc://freenode.net/#openstack-ironic>. This is usually the best place to ask questions and find your way around. IRC stands for Internet Relay Chat and it is a way to chat online in real time. You can ask a question and come back later to read the answer in the log files. Logs for the #openstack-ironic IRC channel are stored at <http://eavesdrop.openstack.org/irclogs/%23openstack-ironic/>.

### Contributor License Agreement

In order to contribute to the python-ironicclient project, you need to have signed OpenStacks contributors agreement.

#### See also:

- <https://docs.openstack.org/infra/manual/developers.html>
- <https://wiki.openstack.org/wiki/CLA>

### Project Hosting Details

**Bug tracker** <https://storyboard.openstack.org/#!/project/959>

**Mailing list (prefix subjects with [ironic] for faster responses)** <http://lists.openstack.org/cgi-bin/mailman/listinfo/openstack-discuss>

**Code Hosting** <https://opendev.org/openstack/python-ironicclient>

**Code Review** <https://review.opendev.org/#/q/status:open+project:openstack/python-ironicclient,n,z>

## 1.4.2 Testing

### Python Guideline Enforcement

All code has to pass the pep8 style guideline to merge into OpenStack, to validate the code against these guidelines you can run:

```
$ tox -e pep8
```

### Unit Testing

It is strongly encouraged to run the unit tests locally under one or more test environments prior to submitting a patch. To run all the recommended environments sequentially and pep8 style guideline run:

```
$ tox
```

You can also selectively pick specific test environments by listing your chosen environments after a -e flag:

```
$ tox -e py3,pep8
```

---

**Note:** Tox sets up virtual environment and installs all necessary dependencies. Sharing the environment with devstack testing is not recommended due to conflicting configuration with system dependencies.

---

## Functional Testing

Functional testing assumes the existence of the script `run_functional.sh` in the `python-ironicclient/tools` directory. The script `run_functional.sh` generates `test.conf` file. To run functional tests just run `./run_functional.sh`.

Also, the `test.conf` file could be created manually or generated from environment variables. It assumes the existence of an openstack cloud installation along with admin credentials. The `test.conf` file lives in `ironicclient/tests/functional/` directory. To run functional tests in that way create `test.conf` manually and run:

```
$ tox -e functional
```

An example `test.conf` file:

```
[functional]
api_version = 1
os_auth_url=http://192.168.0.2:5000/v2.0/
os_username=admin
os_password=admin
os_project_name=admin
```

If you are testing ironic in standalone mode, only the parameters `auth_strategy`, `os_auth_token` and `ironic_url` are required; all others will be ignored.

An example `test.conf` file for standalone host:

```
[functional]
auth_strategy = noauth
os_auth_token = fake
ironic_url = http://10.0.0.2:6385
```

## 1.5 Full Ironic Client Python API Reference

### 1.5.1 ironicclient

#### ironicclient package

##### Subpackages

#### ironicclient.common package

##### Subpackages

#### ironicclient.common.apiclient package

##### Submodules

## ironicclient.common.apiclient.base module

Base utilities to build API operation managers and objects on top of.

**class** `ironicclient.common.apiclient.base.BaseManager` (*client*)

Bases: `ironicclient.common.apiclient.base.HookableMixin`

Basic manager type providing common operations.

Managers interact with a particular type of API (servers, flavors, images, etc.) and provide CRUD operations for them.

**resource\_class** = None

**class** `ironicclient.common.apiclient.base.CrudManager` (*client*)

Bases: `ironicclient.common.apiclient.base.BaseManager`

Base manager class for manipulating entities.

Children of this class are expected to define a *collection\_key* and *key*.

- *collection\_key*: Usually a plural noun by convention (e.g. *entities*); used to refer collections in both URLs (e.g. `/v3/entities`) and JSON objects containing a list of member resources (e.g. `{entities: [{}, {}, {}]}`).
- *key*: Usually a singular noun by convention (e.g. *entity*); used to refer to an individual member of the collection.

**build\_url** (*base\_url=None, \*\*kwargs*)

Builds a resource URL for the given kwargs.

Given an example collection where *collection\_key* = *entities* and *key* = *entity*, the following URLs could be generated.

By default, the URL will represent a collection of entities, e.g.:

```
/entities
```

If kwargs contains an *entity\_id*, then the URL will represent a specific member, e.g.:

```
/entities/{entity_id}
```

**Parameters** *base\_url* if provided, the generated URL will be appended to it

**collection\_key** = None

**create** (*\*\*kwargs*)

**delete** (*\*\*kwargs*)

**find** (*base\_url=None, \*\*kwargs*)

Find a single item with attributes matching *\*\*kwargs*.

**Parameters** *base\_url* if provided, the generated URL will be appended to it

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

**head** (*\*\*kwargs*)

**key** = None

**list** (*base\_url=None, \*\*kwargs*)

List the collection.

**Parameters** **base\_url** if provided, the generated URL will be appended to it

**put** (*base\_url=None, \*\*kwargs*)

Update an element.

**Parameters** **base\_url** if provided, the generated URL will be appended to it

**update** (*\*\*kwargs*)

**class** `ironicclient.common.apiclient.base.Extension` (*name, module*)

Bases: `ironicclient.common.apiclient.base.HookableMixin`

Extension descriptor.

**SUPPORTED\_HOOKS** = ('\_\_pre\_parse\_args\_\_', '\_\_post\_parse\_args\_\_')

**manager\_class** = None

**class** `ironicclient.common.apiclient.base.HookableMixin`

Bases: object

Mixin so classes can register and run hooks.

**classmethod** **add\_hook** (*hook\_type, hook\_func*)

Add a new hook of specified type.

**Parameters**

- **cls** class that registers hooks
- **hook\_type** hook type, e.g., `__pre_parse_args__`
- **hook\_func** hook function

**classmethod** **run\_hooks** (*hook\_type, \*args, \*\*kwargs*)

Run all hooks of specified type.

**Parameters**

- **cls** class that registers hooks
- **hook\_type** hook type, e.g., `__pre_parse_args__`
- **args** args to be passed to every hook function
- **kwargs** kwargs to be passed to every hook function

**class** `ironicclient.common.apiclient.base.ManagerWithFind` (*client*)

Bases: `ironicclient.common.apiclient.base.BaseManager`

Manager with additional `find()/findall()` methods.

**find** (*\*\*kwargs*)

Find a single item with attributes matching `**kwargs`.

This isn't very efficient: it loads the entire list then filters on the Python side.

**findall** (*\*\*kwargs*)

Find all items with attributes matching `**kwargs`.

This isn't very efficient: it loads the entire list then filters on the Python side.

**abstract list()**

**class** `ironicclient.common.apiclient.base.Resource` (*manager*, *info*,  
*loaded=False*)

Bases: `object`

Base class for OpenStack resources (tenant, user, etc.).

This is pretty much just a bag for attributes.

**HUMAN\_ID = False**

**NAME\_ATTR = 'name'**

**get()**

Support for lazy loading details.

Some clients, such as novaclient have the option to lazy load the details, details which can be loaded with this function.

**property human\_id**

Human-readable ID which can be used for bash completion.

**is\_loaded()**

**set\_loaded(val)**

**to\_dict()**

`ironicclient.common.apiclient.base.getid(obj)`

Return id if argument is a Resource.

Abstracts the common pattern of allowing both an object or an objects ID (UUID) as a parameter when dealing with relationships.

## ironicclient.common.apiclient.exceptions module

Exception definitions.

**exception** `ironicclient.common.apiclient.exceptions.AmbiguousEndpoints` (*endpoints=None*)  
Bases: `ironicclient.common.apiclient.exceptions.EndpointException`

Found more than one matching endpoint in Service Catalog.

**exception** `ironicclient.common.apiclient.exceptions.AuthPluginOptionsMissing` (*opt\_name*)  
Bases: `ironicclient.common.apiclient.exceptions.AuthorizationFailure`

Auth plugin misses some options.

**exception** `ironicclient.common.apiclient.exceptions.AuthSystemNotFound` (*auth\_system*)  
Bases: `ironicclient.common.apiclient.exceptions.AuthorizationFailure`

User has specified an AuthSystem that is not installed.

**exception** `ironicclient.common.apiclient.exceptions.AuthorizationFailure`  
Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Cannot authorize API client.



**exception** `ironicclient.common.apiclient.exceptions.BadGateway` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HttpServerError`

HTTP 502 - Bad Gateway.

The server was acting as a gateway or proxy and received an invalid response from the upstream server.

**http\_status = 502**

**message = 'Bad Gateway'**

**exception** `ironicclient.common.apiclient.exceptions.BadRequest` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 400 - Bad Request.

The request cannot be fulfilled due to bad syntax.

**http\_status = 400**

**message = 'Bad Request'**

**exception** `ironicclient.common.apiclient.exceptions.ClientException`

Bases: `Exception`

The base exception class for all exceptions this library raises.

**exception** `ironicclient.common.apiclient.exceptions.CommandError`

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Error in CLI tool.

**exception** `ironicclient.common.apiclient.exceptions.Conflict` (*message=None, de-tails=None, re-sponse=None, re-quest\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 409 - Conflict.

Indicates that the request could not be processed because of conflict in the request, such as an edit conflict.

**http\_status = 409**

**message = 'Conflict'**

**exception** `ironicclient.common.apiclient.exceptions.ConnectionError`

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Cannot connect to API service.

**exception** `ironicclient.common.apiclient.exceptions.ConnectionRefused`

Bases: `ironicclient.common.apiclient.exceptions.ConnectionError`

Connection refused while trying to connect to API service.

**exception** `ironicclient.common.apiclient.exceptions.EndpointException`

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Something is rotten in Service Catalog.

**exception** `ironicclient.common.apiclient.exceptions.EndpointNotFound`

Bases: `ironicclient.common.apiclient.exceptions.EndpointException`

Could not find requested endpoint in Service Catalog.

**exception** `ironicclient.common.apiclient.exceptions.ExpectationFailed` (*message=None, de-*

*tails=None, re-*

*sponse=None, re-*

*quest\_id=None, url=None, method=None, http\_status=None*)

*http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 417 - Expectation Failed.

The server cannot meet the requirements of the Expect request-header field.

**http\_status = 417**

**message = 'Expectation Failed'**

```
exception ironicclient.common.apiclient.exceptions.Forbidden (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 403 - Forbidden.

The request was a valid request, but the server is refusing to respond to it.

```
http_status = 403
```

```
message = 'Forbidden'
```

```
exception ironicclient.common.apiclient.exceptions.GatewayTimeout (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HttpServerError*

HTTP 504 - Gateway Timeout.

The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.

```
http_status = 504
```

```
message = 'Gateway Timeout'
```

```
exception ironicclient.common.apiclient.exceptions.Gone (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 410 - Gone.

Indicates that the resource requested is no longer available and will not be available again.

```
http_status = 410
```

```
message = 'Gone'
```

```
exception ironicclient.common.apiclient.exceptions.HTTPClientError (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HttpError*

Client-side HTTP error.

Exception for cases in which the client seems to have erred.

```
message = 'HTTP Client Error'
```

```
exception ironicclient.common.apiclient.exceptions.HTTPRedirection (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HttpError*

HTTP Redirection.

```
message = 'HTTP Redirection'
```

```
exception ironicclient.common.apiclient.exceptions.HttpError (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.ClientException*

The base exception class for all HTTP exceptions.

```
http_status = 0
```

```
message = 'HTTP Error'
```

**exception** `ironicclient.common.apiclient.exceptions.HttpNotImplemented` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HttpServerError`

HTTP 501 - Not Implemented.

The server either does not recognize the request method, or it lacks the ability to fulfill the request.

**http\_status** = 501

**message** = 'Not Implemented'

**exception** `ironicclient.common.apiclient.exceptions.HttpServerError` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HttpError`

Server-side HTTP error.

Exception for cases in which the server is aware that it has erred or is incapable of performing the request.

**message** = 'HTTP Server Error'

**exception** `ironicclient.common.apiclient.exceptions.HttpVersionNotSupported` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HttpServerError`

HTTP 505 - HttpVersion Not Supported.

The server does not support the HTTP protocol version used in the request.

**http\_status** = 505

```
message = 'HTTP Version Not Supported'
```

```
exception ironicclient.common.apiclient.exceptions.InternalServerError (message=None,
                                details=None,
                                response=None,
                                request_id=None,
                                url=None,
                                method=None,
                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HttpServerError*

HTTP 500 - Internal Server Error.

A generic error message, given when no more specific message is suitable.

```
http_status = 500
```

```
message = 'Internal Server Error'
```

```
exception ironicclient.common.apiclient.exceptions.LengthRequired (message=None,
                                details=None,
                                response=None,
                                request_id=None,
                                url=None,
                                method=None,
                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 411 - Length Required.

The request did not specify the length of its content, which is required by the requested resource.

```
http_status = 411
```

```
message = 'Length Required'
```

```
exception ironicclient.common.apiclient.exceptions.MethodNotAllowed (message=None,
                                details=None,
                                response=None,
                                request_id=None,
                                url=None,
                                method=None,
                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 405 - Method Not Allowed.

A request was made of a resource using a request method not supported by that resource.

```
http_status = 405
```

```
message = 'Method Not Allowed'
```

```
exception ironicclient.common.apiclient.exceptions.MultipleChoices (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

```
Bases: ironicclient.common.apiclient.exceptions.HTTPRedirection
```

HTTP 300 - Multiple Choices.

Indicates multiple options for the resource that the client may follow.

```
http_status = 300
```

```
message = 'Multiple Choices'
```

```
exception ironicclient.common.apiclient.exceptions.NoUniqueMatch
Bases: ironicclient.common.apiclient.exceptions.ClientException
```

Multiple entities found instead of one.

```
exception ironicclient.common.apiclient.exceptions.NotAcceptable (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

```
Bases: ironicclient.common.apiclient.exceptions.HTTPClientError
```

HTTP 406 - Not Acceptable.

The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.

```
http_status = 406
```

```
message = 'Not Acceptable'
```

```
exception ironicclient.common.apiclient.exceptions.NotFound (message=None,
                                                                    de-
                                                                    tails=None,
                                                                    re-
                                                                    sponse=None,
                                                                    re-
                                                                    quest_id=None,
                                                                    url=None,
                                                                    method=None,
                                                                    http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 404 - Not Found.

The requested resource could not be found but may be available again in the future.

**http\_status = 404**

**message = 'Not Found'**

**exception** `ironicclient.common.apiclient.exceptions.PaymentRequired` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 402 - Payment Required.

Reserved for future use.

**http\_status = 402**

**message = 'Payment Required'**

**exception** `ironicclient.common.apiclient.exceptions.PreconditionFailed` (*message=None, details=None, response=None, request\_id=None, url=None, method=None, http\_status=None*)

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 412 - Precondition Failed.

The server does not meet one of the preconditions that the requester put on the request.

**http\_status = 412**

**message = 'Precondition Failed'**



**exception** `ironicclient.common.apiclient.exceptions.ProxyAuthenticationRequired` (*m*

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 407 - Proxy Authentication Required.

The client must first authenticate itself with the proxy.

**http\_status = 407**

**message = 'Proxy Authentication Required'**

**exception** `ironicclient.common.apiclient.exceptions.RequestEntityTooLarge` (*\*args,*  
*\*\*kwargs*)

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 413 - Request Entity Too Large.

The request is larger than the server is willing or able to process.

**http\_status = 413**

**message = 'Request Entity Too Large'**

**exception** `ironicclient.common.apiclient.exceptions.RequestTimeout` (*message=None,*  
*de-*  
*tails=None,*  
*re-*  
*sponse=None,*  
*re-*  
*quest\_id=None,*  
*url=None,*  
*method=None,*  
*http\_status=None*)

Bases: `ironicclient.common.apiclient.exceptions.HTTPClientError`

HTTP 408 - Request Timeout.

The server timed out waiting for the request.

**http\_status = 408**

**message = 'Request Timeout'**

```
exception ironicclient.common.apiclient.exceptions.RequestUriTooLong (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 414 - Request-URI Too Long.

The URI provided was too long for the server to process.

```
http_status = 414
```

```
message = 'Request-URI Too Long'
```

```
exception ironicclient.common.apiclient.exceptions.RequestedRangeNotSatisfiable (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 416 - Requested Range Not Satisfiable.

The client has asked for a portion of the file, but the server cannot supply that portion.

```
http_status = 416
```

```
message = 'Requested Range Not Satisfiable'
```

```
exception ironicclient.common.apiclient.exceptions.ServiceUnavailable (message=None,  
de-  
tails=None,  
re-  
sponse=None,  
re-  
quest_id=None,  
url=None,  
method=None,  
http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HttpServerError*

HTTP 503 - Service Unavailable.

The server is currently unavailable.

```
http_status = 503
```

```
message = 'Service Unavailable'
```

```
exception ironicclient.common.apiclient.exceptions.Unauthorized (message=None,
                                                                de-
                                                                tails=None,
                                                                re-
                                                                sponse=None,
                                                                re-
                                                                quest_id=None,
                                                                url=None,
                                                                method=None,
                                                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 401 - Unauthorized.

Similar to 403 Forbidden, but specifically for use when authentication is required and has failed or has not yet been provided.

```
http_status = 401
```

```
message = 'Unauthorized'
```

```
exception ironicclient.common.apiclient.exceptions.UnprocessableEntity (message=None,
                                                                de-
                                                                tails=None,
                                                                re-
                                                                sponse=None,
                                                                re-
                                                                quest_id=None,
                                                                url=None,
                                                                method=None,
                                                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 422 - Unprocessable Entity.

The request was well-formed but was unable to be followed due to semantic errors.

```
http_status = 422
```

```
message = 'Unprocessable Entity'
```

```
exception ironicclient.common.apiclient.exceptions.UnsupportedMediaType (message=None,
                                                                de-
                                                                tails=None,
                                                                re-
                                                                sponse=None,
                                                                re-
                                                                quest_id=None,
                                                                url=None,
                                                                method=None,
                                                                http_status=None)
```

Bases: *ironicclient.common.apiclient.exceptions.HTTPClientError*

HTTP 415 - Unsupported Media Type.

The request entity has a media type which the server or resource does not support.

```
http_status = 415
```

```
message = 'Unsupported Media Type'
```

```
exception ironicclient.common.apiclient.exceptions.UnsupportedVersion
```

```
Bases: ironicclient.common.apiclient.exceptions.ClientException
```

User is trying to use an unsupported version of the API.

```
exception ironicclient.common.apiclient.exceptions.ValidationError
```

```
Bases: ironicclient.common.apiclient.exceptions.ClientException
```

Error in validation on API client side.

```
ironicclient.common.apiclient.exceptions.from_response(response,  
                                                       method, url)
```

Returns an instance of *HttpError* or subclass based on response.

#### Parameters

- **response** instance of *requests.Response* class
- **method** HTTP method used for request
- **url** URL used for request

## Module contents

### Submodules

#### ironicclient.common.base module

Base utilities to build API operation managers and objects on top of.

```
class ironicclient.common.base.CreateManager(api)
```

```
Bases: ironicclient.common.base.Manager
```

Provides creation operations with a particular API.

```
create(os_ironic_api_version=None, global_request_id=None, **kwargs)
```

Create a resource based on a kwargs dictionary of attributes.

#### Parameters

- **kwargs** A dictionary containing the attributes of the resource that will be created.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Raises** *exc.InvalidAttribute* For invalid attributes that are not needed to create the resource.

```
class ironicclient.common.base.Manager(api)
```

```
Bases: object
```

Provides CRUD operations with a particular API.

### **abstract property resource\_class**

The resource class

**class** `ironicclient.common.base.Resource` (*manager, info, loaded=False*)

Bases: `ironicclient.common.apiclient.base.Resource`

Represents a particular instance of an object (tenant, user, etc).

This is pretty much just a bag for attributes.

**to\_dict** ()

`ironicclient.common.base.get_id` (*obj*)

Wrapper to get objects ID.

Abstracts the common pattern of allowing both an object or an objects ID (UUID) as a parameter when dealing with relationships.

### **ironicclient.common.filecache module**

`ironicclient.common.filecache.retrieve_data` (*host, port, expiry=None*)

Retrieve the version stored for an ironic host, if its not stale.

Check to see if there is valid cached data for the host/port combination and return that if it isnt stale.

param host: The host that we need to retrieve data for  
 param port: The port on the host that we need to retrieve data for  
 param expiry: The age in seconds before cached data is deemed invalid

`ironicclient.common.filecache.save_data` (*host, port, data*)

Save data for a particular host in the appropriate cache dir.

param host: The host that we need to save data for  
 param port: The port on the host that we need to save data for  
 param data: The data we want saved

### **ironicclient.common.http module**

**class** `ironicclient.common.http.SessionClient` (*os\_ironic\_api\_version, api\_version\_select\_state, max\_retries, retry\_interval, \*\*kwargs*)

Bases: `ironicclient.common.http.VersionNegotiationMixin`,  
`keystoneauth1.adapter.LegacyJsonAdapter`

HTTP client based on Keystone client session.

**json\_request** (*method, url, \*\*kwargs*)

**raw\_request** (*method, url, \*\*kwargs*)

**class** `ironicclient.common.http.VersionNegotiationMixin`

Bases: `object`

**negotiate\_version** (*conn, resp*)

Negotiate the server version

Assumption: Called after receiving a 406 error when doing a request.

### Parameters

- **conn** A connection object
- **resp** The response object from http request

`ironicclient.common.http.get_server(url)`

Extract and return the server & port.

`ironicclient.common.http.with_retries(func)`

Wrapper for `_http_request` adding support for retries.

## ironicclient.common.i18n module

## ironicclient.common.utils module

**class** `ironicclient.common.utils.HelpFormatter` (*prog, indent\_increment=2, max\_help\_position=24, width=None*)

Bases: `argparse.HelpFormatter`

**start\_section** (*heading*)

`ironicclient.common.utils.args_array_to_dict` (*kwargs, key\_to\_convert*)

Convert the value in a dictionary entry to a dictionary.

From the `kwargs` dictionary, converts the value of the `key_to_convert` entry from a list of key-value pairs to a dictionary.

### Parameters

- **kwargs** a dictionary
- **key\_to\_convert** the key (in `kwargs`), whose value is expected to be a list of key=value strings. This value will be converted to a dictionary.

**Returns** `kwargs`, the (modified) dictionary

`ironicclient.common.utils.args_array_to_patch` (*op, attributes*)

`ironicclient.common.utils.bool_argument_value` (*arg\_name, bool\_str, strict=True, default=False*)

Returns the Boolean represented by `bool_str`.

Returns the Boolean value for the argument named `arg_name`. The value is represented by the string `bool_str`. If the string is an invalid Boolean string: if `strict` is `True`, a `CommandError` exception is raised; otherwise the default value is returned.

### Parameters

- **arg\_name** The name of the argument
- **bool\_str** The string representing a Boolean value
- **strict** Used if the string is invalid. If `True`, raises an exception. If `False`, returns the default value.
- **default** The default value to return if the string is invalid and not strict

**Returns** the Boolean value represented by `bool_str` or the default value if `bool_str` is invalid and `strict` is `False`

**Raises** *CommandError* if `bool_str` is an invalid Boolean string

`ironicclient.common.utils.check_empty_arg` (*arg, arg\_descriptor*)

`ironicclient.common.utils.check_for_invalid_fields` (*fields, valid\_fields*)

Check for invalid fields.

**Parameters**

- **fields** A list of fields specified by the user.
- **valid\_fields** A list of valid fields.

**Raises** *CommandError* If invalid fields were specified by the user.

`ironicclient.common.utils.common_filters` (*marker=None, limit=None, sort\_key=None, sort\_dir=None, fields=None, detail=False*)

Generate common filters for any list request.

**Parameters**

- **marker** entity ID from which to start returning entities.
- **limit** maximum number of entities to return.
- **sort\_key** field to use for sorting.
- **sort\_dir** direction of sorting: asc or desc.
- **fields** a list with a specified set of fields of the resource to be returned.
- **detail** Boolean, True to return detailed information. This parameter can be used for resources which accept detail as a URL parameter.

**Returns** list of string filters.

`ironicclient.common.utils.common_params_for_list` (*args, fields, field\_labels*)

Generate params dict that is common for every list command.

**Parameters**

- **args** arguments from command line.
- **fields** possible fields for sorting.
- **field\_labels** possible field labels for sorting.

**Returns** a dict with params to pass to the client method.

`ironicclient.common.utils.convert_list_props_to_comma_separated` (*data, props=None*)

Convert the list-type properties to comma-separated strings

**Parameters**

- **data** the input dict object.
- **props** the properties whose values will be converted. Default to None to convert all list-type properties of the input.

**Returns** the result dict instance.

`ironicclient.common.utils.define_command(subparsers, command, callback, cmd_mapper)`

Define a command in the subparsers collection.

**Parameters**

- **subparsers** subparsers collection where the command will go
- **command** command name
- **callback** function that will be used to process the command

`ironicclient.common.utils.define_commands_from_module(subparsers, command_module, cmd_mapper)`

Add *do\_* methods in a module and add as commands into a subparsers.

`ironicclient.common.utils.get_from_stdin(info_desc)`

Read information from stdin.

**Parameters** **info\_desc** A string description of the desired information

**Raises** `InvalidAttribute` if there was a problem reading from stdin

**Returns** the string that was read from stdin

`ironicclient.common.utils.handle_json_arg(json_arg, info_desc)`

Read a JSON argument from stdin, file or string.

**Parameters**

- **json\_arg** May be a file name containing the JSON, a JSON string, or - indicating that the argument should be read from standard input.
- **info\_desc** A string description of the desired information

**Returns** A list or dictionary parsed from JSON.

**Raises** `InvalidAttribute` if the argument cannot be parsed.

`ironicclient.common.utils.handle_json_or_file_arg(json_arg)`

Attempts to read JSON argument from file or string.

**Parameters** **json\_arg** May be a file name containing the JSON, or a JSON string.

**Returns** A list or dictionary parsed from JSON.

**Raises** `InvalidAttribute` if the argument cannot be parsed.

`ironicclient.common.utils.key_value_pairs_to_dict(key_value_pairs)`

Convert a list of key-value pairs to a dictionary.

**Parameters** **key\_value\_pairs** a list of strings, each string is in the form <key>=<value>

**Returns** a dictionary, possibly empty

`ironicclient.common.utils.make_configdrive(path)`

Make the config drive file.

**Parameters** **path** The directory containing the config drive files.

**Returns** A gzipped and base64 encoded configdrive string.



`ironicclient.common.utils.poll` (*timeout, poll\_interval, poll\_delay\_function, timeout\_message*)

`ironicclient.common.utils.split_and_deserialize` (*string*)

Split and try to JSON deserialize a string.

Gets a string with the KEY=VALUE format, split it (using = as the separator) and try to JSON deserialize the VALUE.

**Returns** A tuple of (key, value).

`ironicclient.common.utils.tempdir` (*\*args, \*\*kwargs*)

## Module contents

### ironicclient.osc package

#### Subpackages

#### ironicclient.osc.v1 package

#### Submodules

#### ironicclient.osc.v1.baremetal\_allocation module

**class** `ironicclient.osc.v1.baremetal_allocation.CreateBaremetalAllocation` (*app, app\_args, cmd\_name*)

Bases: `osc_lib.command.command.ShowOne`

Create a new baremetal allocation.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_allocation.CreateBaremetalAllocation`

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocation` (*app, app\_args, cmd\_name*)

Bases: `osc_lib.command.command.Command`

Unregister baremetal allocation(s).

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocation`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

```

class ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocation (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Lister

    List baremetal allocations.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocati

    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_allocation.SetBaremetalAllocation (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Command

    Set baremetal allocation properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_allocation.SetBaremetalAllocatio

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocation (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.ShowOne

    Show baremetal allocation details.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocati

    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

class ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocation (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Command

    Unset baremetal allocation properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocat

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

```

## ironicclient.osc.v1.baremetal\_chassis module

```

class ironicclient.osc.v1.baremetal_chassis.CreateBaremetalChassis (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.ShowOne

    Create a new chassis.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_chassis.CreateBaremetalChassis (
    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

class ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassis (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Command

    Delete a chassis.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassis (
    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassis (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Lister

    List the chassis.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassis (WA
    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_chassis.SetBaremetalChassis (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Command

    Set chassis properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_chassis.SetBaremetalChassis (WAR

```

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

**class** `ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassis` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.ShowOne`

Show chassis details.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassis (WARNING)`

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassis` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.Command`

Unset chassis properties.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassis (WARNING)`

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

## ironicclient.osc.v1.baremetal\_conductor module

**class** `ironicclient.osc.v1.baremetal_conductor.ListBaremetalConductor` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.Lister`

List baremetal conductors

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_conductor.ListBaremetalNode (WARNING)`

**take\_action** (*parsed\_args*)  
 Return a tuple containing the column names and an iterable containing the data to be listed.

**class** `ironicclient.osc.v1.baremetal_conductor.ShowBaremetalConductor` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.ShowOne`

Show baremetal conductor details

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_conductor.ShowBaremetalConductor`

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

### ironicclient.osc.v1.baremetal\_create module

```
class ironicclient.osc.v1.baremetal_create.CreateBaremetal (app,
                                                         app_args,
                                                         cmd_name=None)
```

Bases: `osc_lib.command.command.Command`

Create resources from files

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_create.CreateBaremetal (WARNING)`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

### ironicclient.osc.v1.baremetal\_deploy\_template module

```
class ironicclient.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplat
```

Bases: `osc_lib.command.command.ShowOne`

Create a new deploy template

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.CreateBaremetalD`

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

```
class ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplat
```

Bases: `osc_lib.command.command.Command`

Delete deploy template(s).

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalD`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplate (
```

Bases: `osc_lib.command.command.Lister`

List baremetal deploy templates.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDep`

**take\_action** (*parsed\_args*)

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate (a
```

Bases: `osc_lib.command.command.Command`

Set baremetal deploy template properties.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDep`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate (
```

Bases: `osc_lib.command.command.ShowOne`

Show baremetal deploy template details.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDep`

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

```
class ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate
```

Bases: `osc_lib.command.command.Command`

Unset baremetal deploy template properties.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDe`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

### ironicclient.osc.v1.baremetal\_driver module

```
class ironicclient.osc.v1.baremetal_driver.ListBaremetalDriver (app,
                                                             app_args,
                                                             cmd_name=None)

    Bases: osc_lib.command.command.Lister

    List the enabled drivers.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_driver.ListBaremetalDriver (WARN

    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverProperty (app,
                                                                           app_args,
                                                                           cmd_name=None)

    Bases: osc_lib.command.command.Lister

    List the driver properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverProper

    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverRaidProperty (app,
                                                                               app_ar,
                                                                               cmd_na)

    Bases: osc_lib.command.command.Lister

    List a drivers RAID logical disk properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverRaidPr

    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_driver.PassthruCallBaremetalDriver (app,
                                                                           app_args,
                                                                           cmd_name=None)

    Bases: osc_lib.command.command.ShowOne

    Call a vendor passthru method for a driver.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_driver.PassthruCallBaremetalDriv
```

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriver` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.Lister`

List available vendor passthru methods for a driver.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriver`

**take\_action** (*parsed\_args*)  
 Return a tuple containing the column names and an iterable containing the data to be listed.

**class** `ironicclient.osc.v1.baremetal_driver.ShowBaremetalDriver` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.ShowOne`

Show information about a driver.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_driver.ShowBaremetalDriver` (WARNING)

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

## ironicclient.osc.v1.baremetal\_node module

**class** `ironicclient.osc.v1.baremetal_node.AbortBaremetalNode` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode`

Set provision state of baremetal node to abort

**PROVISION\_STATE** = 'abort'

**log** = `<Logger ironicclient.osc.v1.baremetal_node.AbortBaremetalNode` (WARNING)

**class** `ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNode` (*app*,  
*app\_args*,  
*cmd\_name=None*)

Bases: `osc_lib.command.command.Command`

Add traits to a node.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNode` (WARNING)

**take\_action** (*parsed\_args*)  
 Override to do something useful.



The returned value will be returned by the program.

```

class ironicclient.osc.v1.baremetal_node.AdoptBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)
    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait
    Set provision state of baremetal node to adopt
    PROVISION_STATE = 'adopt'
    log = <Logger ironicclient.osc.v1.baremetal_node.AdoptBaremetalNode (WARNING)

class ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode (app,
                                                                           app_args,
                                                                           cmd_name=None)
    Bases: osc_lib.command.command.ShowOne
    Show a specific BIOS setting for a node.
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode
    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

class ironicclient.osc.v1.baremetal_node.BootdeviceSetBaremetalNode (app,
                                                                           app_args,
                                                                           cmd_name=None)
    Bases: osc_lib.command.command.Command
    Set the boot device for a node
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.BootdeviceSetBaremetalNode
    take_action (parsed_args)
        Override to do something useful.
        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.BootdeviceShowBaremetalNode (app,
                                                                           app_args,
                                                                           cmd_name=None)
    Bases: osc_lib.command.command.ShowOne
    Show the boot device information for a node
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.BootdeviceShowBaremetalNode
    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

```

```

class ironicclient.osc.v1.baremetal_node.CleanBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)
    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait
    Set provision state of baremetal node to clean
    PROVISION_STATE = 'clean'
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.CleanBaremetalNode (WARNING)

class ironicclient.osc.v1.baremetal_node.ConsoleDisableBaremetalNode (app,
                                                                       app_args,
                                                                       cmd_name=None)
    Bases: osc_lib.command.command.Command
    Disable console access for a node
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ConsoleDisableBaremetalNode
    take_action (parsed_args)
        Override to do something useful.
        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.ConsoleEnableBaremetalNode (app,
                                                                       app_args,
                                                                       cmd_name=None)
    Bases: osc_lib.command.command.Command
    Enable console access for a node
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ConsoleEnableBaremetalNode
    take_action (parsed_args)
        Override to do something useful.
        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.ConsoleShowBaremetalNode (app,
                                                                     app_args,
                                                                     cmd_name=None)
    Bases: osc_lib.command.command.ShowOne
    Show console information for a node
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ConsoleShowBaremetalNode (W
    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

```

```

class ironicclient.osc.v1.baremetal_node.CreateBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)

    Bases: osc_lib.command.command.ShowOne

    Register a new node with the baremetal service

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.CreateBaremetalNode (WARNING)

    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

class ironicclient.osc.v1.baremetal_node.DeleteBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)

    Bases: osc_lib.command.command.Command

    Unregister baremetal node(s)

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.DeleteBaremetalNode (WARNING)

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.DeployBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait

    Set provision state of baremetal node to deploy

    PROVISION_STATE = 'active'

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.DeployBaremetalNode (WARNING)

class ironicclient.osc.v1.baremetal_node.InjectNmiBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)

    Bases: osc_lib.command.command.Command

    Inject NMI to baremetal node

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.InjectNmiBaremetalNode (WARNING)

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

```

```

class ironicclient.osc.v1.baremetal_node.InspectBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait
    Set provision state of baremetal node to inspect
    PROVISION_STATE = 'inspect'
    log = <Logger ironicclient.osc.v1.baremetal_node.InspectBaremetalNode (WARNING)>

class ironicclient.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode (app,
                                                                        app_args,
                                                                        cmd_name=None)
    Bases: osc_lib.command.command.Lister
    List a nodes BIOS settings.
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode (WARNING)>
    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_node.ListBaremetalNode (app,
                                                            app_args,
                                                            cmd_name=None)
    Bases: osc_lib.command.command.Lister
    List baremetal nodes
    PROVISION_STATES = ['active', 'deleted', 'rebuild', 'inspect', 'provide', 'm
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ListBaremetalNode (WARNING)>
    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_node.ListTraitsBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)
    Bases: osc_lib.command.command.Lister
    List a nodes traits.
    get_parser (prog_name)
        Return an argparse.ArgumentParser.
    log = <Logger ironicclient.osc.v1.baremetal_node.ListTraitsBaremetalNode (WARNING)>
    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_node.MaintenanceSetBaremetalNode (app,
                                                                        app_args,
                                                                        cmd_name=None)
    Bases: osc_lib.command.command.Command

```

Set baremetal node to maintenance mode

**get\_parser** (*prog\_name*)  
Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.MaintenanceSetBaremetalNode`

**take\_action** (*parsed\_args*)  
Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)
```

Bases: `osc_lib.command.command.Command`

Unset baremetal node from maintenance mode

**get\_parser** (*prog\_name*)  
Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode`

**take\_action** (*parsed\_args*)  
Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ManageBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
```

Bases: `ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait`

Set provision state of baremetal node to manage

**PROVISION\_STATE** = 'manage'

**log** = `<Logger ironicclient.osc.v1.baremetal_node.ManageBaremetalNode (WARNIN`

```
class ironicclient.osc.v1.baremetal_node.PassthruCallBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)
```

Bases: `osc_lib.command.command.Command`

Call a vendor passthru method for a node

**get\_parser** (*prog\_name*)  
Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.PassthruCallBaremetalNode (W`

**take\_action** (*parsed\_args*)  
Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.PassthruListBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)
```

Bases: `osc_lib.command.command.Lister`

List vendor passthru methods for a node

```
get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.PassthruListBaremetalNode (WARNING)

take_action (parsed_args)
    Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_node.PowerBaremetalNode (app,
                                                         app_args,
                                                         cmd_name=None)

    Bases: osc_lib.command.command.Command

    Base power state class, for setting the power of a node

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.PowerBaremetalNode (WARNING)

take_action (parsed_args)
    Override to do something useful.

    The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.PowerOffBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.PowerBaremetalNode

    Power off a node

POWER_STATE = 'off'

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.PowerOffBaremetalNode (WARNING)

class ironicclient.osc.v1.baremetal_node.PowerOnBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.PowerBaremetalNode

    Power on a node

POWER_STATE = 'on'

log = <Logger ironicclient.osc.v1.baremetal_node.PowerOnBaremetalNode (WARNING)

class ironicclient.osc.v1.baremetal_node.ProvideBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait

    Set provision state of baremetal node to provide

PROVISION_STATE = 'provide'

log = <Logger ironicclient.osc.v1.baremetal_node.ProvideBaremetalNode (WARNING)
```

```

class ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: osc_lib.command.command.Command

    Base provision state class

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait (app,
                                                                    app_args,
                                                                    cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode

    Provision state class adding wait flag.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait (WAR

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.RebootBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: osc_lib.command.command.Command

    Reboot baremetal node

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_node.RebootBaremetalNode (WARNIN

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.RebuildBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait

    Set provision state of baremetal node to rebuild

    PROVISION_STATE = 'rebuild'

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

```

```

    log = <Logger ironicclient.osc.v1.baremetal_node.RebuildBaremetalNode (WARNING)>
class ironicclient.osc.v1.baremetal_node.RemoveTraitBaremetalNode (app,
                                                                    app_args,
                                                                    cmd_name=None)

Bases: osc_lib.command.command.Command

Remove trait(s) from a node.

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.RemoveTraitBaremetalNode (WARNING)>

take_action (parsed_args)
    Override to do something useful.

    The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.RescueBaremetalNode (app,
                                                             app_args,
                                                             cmd_name=None)

Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait

Set provision state of baremetal node to rescue

PROVISION_STATE = 'rescue'

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.RescueBaremetalNode (WARNING)>

class ironicclient.osc.v1.baremetal_node.SetBaremetalNode (app,
                                                           app_args,
                                                           cmd_name=None)

Bases: osc_lib.command.command.Command

Set baremetal properties

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.SetBaremetalNode (WARNING)>

take_action (parsed_args)
    Override to do something useful.

    The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_node.ShowBaremetalNode (app,
                                                            app_args,
                                                            cmd_name=None)

Bases: osc_lib.command.command.ShowOne

Show baremetal node details

get_parser (prog_name)
    Return an argparse.ArgumentParser.

log = <Logger ironicclient.osc.v1.baremetal_node.ShowBaremetalNode (WARNING)>

```



**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

```
class ironicclient.osc.v1.baremetal_node.UndeployBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait
```

Set provision state of baremetal node to deleted

**PROVISION\_STATE** = 'deleted'

**log** = <Logger ironicclient.osc.v1.baremetal\_node.UndeployBaremetalNode (WARN

```
class ironicclient.osc.v1.baremetal_node.UnrescueBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
Bases: ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait
```

Set provision state of baremetal node to unrescue

**PROVISION\_STATE** = 'unrescue'

**log** = <Logger ironicclient.osc.v1.baremetal\_node.UnrescueBaremetalNode (WARN

```
class ironicclient.osc.v1.baremetal_node.UnsetBaremetalNode (app,
                                                            app_args,
                                                            cmd_name=None)
Bases: osc_lib.command.command.Command
```

Unset baremetal properties

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = <Logger ironicclient.osc.v1.baremetal\_node.UnsetBaremetalNode (WARNING

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ValidateBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
Bases: osc_lib.command.command.Lister
```

Validate a nodes driver interfaces

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = <Logger ironicclient.osc.v1.baremetal\_node.ValidateBaremetalNode (WARN

**take\_action** (*parsed\_args*)

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.VifAttachBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
Bases: osc_lib.command.command.Command
```

Attach VIF to a given node

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.VifAttachBaremetalNode (WARN`

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.VifDetachBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
```

Bases: `osc_lib.command.command.Command`

Detach VIF from a given node

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.VifDetachBaremetalNode (WARN`

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.VifListBaremetalNode (app,
                                                                app_args,
                                                                cmd_name=None)
```

Bases: `osc_lib.command.command.Lister`

Show attached VIFs for a node

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_node.VifListBaremetalNode (WARNIN`

**take\_action** (*parsed\_args*)  
 Return a tuple containing the column names and an iterable containing the data to be listed.

## ironicclient.osc.v1.baremetal\_port module

```
class ironicclient.osc.v1.baremetal_port.CreateBaremetalPort (app,
                                                                app_args,
                                                                cmd_name=None)
```

Bases: `osc_lib.command.command.ShowOne`

Create a new port

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_port.CreateBaremetalPort (WARNIN`

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

```

class ironicclient.osc.v1.baremetal_port.DeleteBaremetalPort (app,
                                                                app_args,
                                                                cmd_name=None)

    Bases: osc_lib.command.command.Command

    Delete port(s).

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_port.DeleteBaremetalPort (WARNING)>

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_port.ListBaremetalPort (app,
                                                            app_args,
                                                            cmd_name=None)

    Bases: osc_lib.command.command.Lister

    List baremetal ports.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_port.ListBaremetalPort (WARNING)>

    take_action (parsed_args)
        Return a tuple containing the column names and an iterable containing the data to be listed.

class ironicclient.osc.v1.baremetal_port.SetBaremetalPort (app,
                                                            app_args,
                                                            cmd_name=None)

    Bases: osc_lib.command.command.Command

    Set baremetal port properties.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_port.SetBaremetalPort (WARNING)>

    take_action (parsed_args)
        Override to do something useful.

        The returned value will be returned by the program.

class ironicclient.osc.v1.baremetal_port.ShowBaremetalPort (app,
                                                            app_args,
                                                            cmd_name=None)

    Bases: osc_lib.command.command.ShowOne

    Show baremetal port details.

    get_parser (prog_name)
        Return an argparse.ArgumentParser.

    log = <Logger ironicclient.osc.v1.baremetal_port.ShowBaremetalPort (WARNING)>

    take_action (parsed_args)
        Return a two-part tuple with a tuple of column names and a tuple of values.

```

```
class ironicclient.osc.v1.baremetal_port.UnsetBaremetalPort (app,  
                                                             app_args,  
                                                             cmd_name=None)  
  
    Bases: osc_lib.command.command.Command  
  
    Unset baremetal port properties.  
  
    get_parser (prog_name)  
        Return an argparse.ArgumentParser.  
  
    log = <Logger ironicclient.osc.v1.baremetal_port.UnsetBaremetalPort (WARNING)  
  
    take_action (parsed_args)  
        Override to do something useful.  
  
        The returned value will be returned by the program.
```

### ironicclient.osc.v1.baremetal\_portgroup module

```
class ironicclient.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup (app,  
                                                                           app_args,  
                                                                           cmd_name=None)  
  
    Bases: osc_lib.command.command.ShowOne  
  
    Create a new baremetal port group.  
  
    get_parser (prog_name)  
        Return an argparse.ArgumentParser.  
  
    log = <Logger ironicclient.osc.v1.baremetal_portgroup.CreateBaremetalPortGro  
  
    take_action (parsed_args)  
        Return a two-part tuple with a tuple of column names and a tuple of values.
```

```
class ironicclient.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup (app,  
                                                                           app_args,  
                                                                           cmd_name=None)  
  
    Bases: osc_lib.command.command.Command  
  
    Unregister baremetal port group(s).  
  
    get_parser (prog_name)  
        Return an argparse.ArgumentParser.  
  
    log = <Logger ironicclient.osc.v1.baremetal_portgroup.DeleteBaremetalPortGro  
  
    take_action (parsed_args)  
        Override to do something useful.  
  
        The returned value will be returned by the program.
```

```
class ironicclient.osc.v1.baremetal_portgroup.ListBaremetalPortGroup (app,  
                                                                           app_args,  
                                                                           cmd_name=None)  
  
    Bases: osc_lib.command.command.Lister  
  
    List baremetal port groups.  
  
    get_parser (prog_name)  
        Return an argparse.ArgumentParser.
```

**log** = <Logger `ironicclient.osc.v1.baremetal_portgroup.ListBaremetalPortGroup`

**take\_action** (*parsed\_args*)

Return a tuple containing the column names and an iterable containing the data to be listed.

**class** `ironicclient.osc.v1.baremetal_portgroup.SetBaremetalPortGroup` (*app*,  
*app\_args*,  
*cmd\_name=None*

Bases: `osc_lib.command.command.Command`

Set baremetal port group properties.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = <Logger `ironicclient.osc.v1.baremetal_portgroup.SetBaremetalPortGroup`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

**class** `ironicclient.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup` (*app*,  
*app\_args*,  
*cmd\_name=None*

Bases: `osc_lib.command.command.ShowOne`

Show baremetal port group details.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = <Logger `ironicclient.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup`

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup` (*app*,  
*app\_args*,  
*cmd\_name=None*

Bases: `osc_lib.command.command.Command`

Unset baremetal port group properties.

**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = <Logger `ironicclient.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

## ironicclient.osc.v1.baremetal\_volume\_connector module

**class** ironicclient.osc.v1.baremetal\_volume\_connector.CreateBaremetalVolumeConnec

Bases: osc\_lib.command.command.ShowOne

Create a new baremetal volume connector.

**get\_parser** (*prog\_name*)

Return an argparse.ArgumentParser.

**log** = <Logger ironicclient.osc.v1.baremetal\_volume\_connector.CreateBaremetal

**take\_action** (*parsed\_args*)

Return a two-part tuple with a tuple of column names and a tuple of values.

**class** ironicclient.osc.v1.baremetal\_volume\_connector.DeleteBaremetalVolumeConnec

Bases: osc\_lib.command.command.Command

Unregister baremetal volume connector(s).

**get\_parser** (*prog\_name*)

Return an argparse.ArgumentParser.

**log** = <Logger ironicclient.osc.v1.baremetal\_volume\_connector.DeleteBaremetal

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

**class** ironicclient.osc.v1.baremetal\_volume\_connector.ListBaremetalVolumeConnecto

Bases: osc\_lib.command.command.Lister

List baremetal volume connectors.

**get\_parser** (*prog\_name*)

Return an argparse.ArgumentParser.

**log** = <Logger ironicclient.osc.v1.baremetal\_volume\_connector.ListBaremetalVo

**take\_action** (*parsed\_args*)

Return a tuple containing the column names and an iterable containing the data to be listed.

**class** ironicclient.osc.v1.baremetal\_volume\_connector.SetBaremetalVolumeConnector

Bases: osc\_lib.command.command.Command

Set baremetal volume connector properties.

**get\_parser** (*prog\_name*)

Return an argparse.ArgumentParser.

**log** = <Logger ironicclient.osc.v1.baremetal\_volume\_connector.SetBaremetalVol

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

**class** `ironicclient.osc.v1.baremetal_volume_connector.ShowBaremetalVolumeConnector`

Bases: `osc_lib.command.command.ShowOne`

Show baremetal volume connector details.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_connector.ShowBaremetalVolumeConnector>`

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_volume_connector.UnsetBaremetalVolumeConnector`

Bases: `osc_lib.command.command.Command`

Unset baremetal volume connector properties.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_connector.UnsetBaremetalVolumeConnector>`

**take\_action** (*parsed\_args*)  
 Override to do something useful.  
 The returned value will be returned by the program.

## ironicclient.osc.v1.baremetal\_volume\_target module

**class** `ironicclient.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget` (*app*, *app*, *cm*)

Bases: `osc_lib.command.command.ShowOne`

Create a new baremetal volume target.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget>`

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget` (*app*, *app*, *cm*)

Bases: `osc_lib.command.command.Command`

Unregister baremetal volume target(s).

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget>`

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

**class** `ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget` (*app*,  
*app\_args*  
*cmd\_name*)

Bases: `osc_lib.command.command.Lister`

List baremetal volume targets.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget>`

**take\_action** (*parsed\_args*)  
 Return a tuple containing the column names and an iterable containing the data to be listed.

**class** `ironicclient.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget` (*app*,  
*app\_args*  
*cmd\_name*)

Bases: `osc_lib.command.command.Command`

Set baremetal volume target properties.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget>`

**take\_action** (*parsed\_args*)  
 Override to do something useful.

The returned value will be returned by the program.

**class** `ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget` (*app*,  
*app\_args*  
*cmd\_name*)

Bases: `osc_lib.command.command.ShowOne`

Show baremetal volume target details.

**get\_parser** (*prog\_name*)  
 Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget>`

**take\_action** (*parsed\_args*)  
 Return a two-part tuple with a tuple of column names and a tuple of values.

**class** `ironicclient.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget` (*app*,  
*app\_args*  
*cmd\_name*)

Bases: `osc_lib.command.command.Command`

Unset baremetal volume target properties.



**get\_parser** (*prog\_name*)

Return an `argparse.ArgumentParser`.

**log** = `<Logger ironicclient.osc.v1.baremetal_volume_targetUnsetBaremetalVolum`

**take\_action** (*parsed\_args*)

Override to do something useful.

The returned value will be returned by the program.

## Module contents

### Submodules

#### ironicclient.osc.plugin module

OpenStackClient plugin for Bare Metal service.

```
class ironicclient.osc.plugin.ReplaceLatestVersion (option_strings, dest,
                                                    nargs=None,
                                                    const=None,
                                                    default=None,
                                                    type=None,
                                                    choices=None,
                                                    required=False,
                                                    help=None,
                                                    metavar=None)
```

Bases: `argparse.Action`

Replaces *latest* keyword by last known version.

OSC cannot accept the literal latest as a supported API version as it breaks the major version detection (OSC tries to load configuration options from `setuptools.entrypoint.openstack.baremetal.vlatest`). This action replaces latest with the latest known version, and sets the global `OS_BAREMETAL_API_LATEST` flag appropriately.

```
ironicclient.osc.plugin.build_option_parser (parser)
```

Hook to add global options.

```
ironicclient.osc.plugin.make_client (instance)
```

Returns a baremetal service client.

## Module contents

### ironicclient.v1 package

#### Submodules

#### ironicclient.v1.allocation module

```
class ironicclient.v1.allocation.Allocation (manager, info, loaded=False)
```

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.allocation.AllocationManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**delete** (*allocation\_id, os\_ironic\_api\_version=None, global\_request\_id=None*)

Delete the Allocation.

#### Parameters

- **allocation\_id** The UUID or name of an allocation.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**get** (*allocation\_id, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Get an allocation with the specified identifier.

#### Parameters

- **allocation\_id** The UUID or name of an allocation.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** an `Allocation` object.

**list** (*resource\_class=None, state=None, node=None, limit=None, marker=None, sort\_key=None, sort\_dir=None, fields=None, owner=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Retrieve a list of allocations.

#### Parameters

- **resource\_class** Optional, get allocations with this resource class.
- **state** Optional, get allocations in this state. One of `allocating`, `active` or `error`.
- **node** UUID or name of the node of the allocation.
- **marker** Optional, the UUID of an allocation, eg the last allocation from a previous result set. Return the next result set.
- **limit**

**The maximum number of results to return per** request, if:

- 1) `limit > 0`, the maximum number of allocations to return.
- 2) `limit == 0`, return the entire list of allocations.
- 3) `limit == None`, the number of items returned respect the maximum imposed by the Ironic API (see Ironics `api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either asc (the default) or desc.
- **fields** Optional, a list with a specified set of fields of the resource to be returned.
- **owner** Optional, project that owns the allocation.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of allocations.

**Raises** InvalidAttribute if a subset of fields is requested with detail option set.

#### **resource\_class**

alias of *Allocation*

**update** (*allocation\_id, patch, os\_ironic\_api\_version=None, global\_request\_id=None*)

Updates the Allocation. Only name and extra field are allowed.

#### **Parameters**

- **allocation\_id** The UUID or name of an allocation.
- **patch** a json PATCH document to apply to this allocation.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**wait** (*allocation\_id, timeout=0, poll\_interval=1, poll\_delay\_function=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Wait for the Allocation to become active.

#### **Parameters**

- **timeout** timeout in seconds, no timeout if 0.
- **poll\_interval** interval in seconds between polls.
- **poll\_delay\_function** function to use to wait between polls (defaults to time.sleep). Should take one argument - delay time in seconds. Any exceptions raised inside it will abort the wait.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** updated *Allocation* object.

**Raises** StateTransitionFailed if allocation reaches the error state.

**Raises** StateTransitionTimeout on timeout.

## ironicclient.v1.chassis module

**class** `ironicclient.v1.chassis.Chassis` (*manager, info, loaded=False*)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.chassis.ChassisManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**delete** (*chassis\_id, os\_ironic\_api\_version=None, global\_request\_id=None*)

**get** (*chassis\_id, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

**list** (*marker=None, limit=None, sort\_key=None, sort\_dir=None, detail=False, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Retrieve a list of chassis.

### Parameters

- **marker** Optional, the UUID of a chassis, eg the last chassis from a previous result set. Return the next result set.
- **limit**  
The maximum number of results to return per request, if:
  - 1) `limit > 0`, the maximum number of chassis to return.
  - 2) `limit == 0`, return the entire list of chassis.
  - 3) `limit` param is NOT specified (`None`), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about chassis.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of chassis.

**list\_nodes** (*chassis\_id, marker=None, limit=None, sort\_key=None, sort\_dir=None, detail=False, fields=None, associated=None, maintenance=None, provision\_state=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

List all the nodes for a given chassis.

### Parameters

- **chassis\_id** The UUID of the chassis.

- **marker** Optional, the UUID of a node, eg the last node from a previous result set. Return the next result set.
- **limit**  
**The maximum number of results to return per request, if:**
  - 1) `limit > 0`, the maximum number of nodes to return.
  - 2) `limit == 0`, return the entire list of nodes.
  - 3) `limit` param is NOT specified (`None`), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about nodes.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **associated** Optional. Either a Boolean or a string representation of a Boolean that indicates whether to return a list of associated (`True` or `True`) or unassociated (`False` or `False`) nodes.
- **maintenance** Optional. Either a Boolean or a string representation of a Boolean that indicates whether to return nodes in maintenance mode (`True` or `True`), or not in maintenance mode (`False` or `False`).
- **provision\_state** Optional. String value to get only nodes in that provision state.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of nodes.

**resource\_class**

alias of *Chassis*

**update** (*chassis\_id*, *patch*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

## ironicclient.v1.client module

**class** `ironicclient.v1.client.Client` (*endpoint\_override=None*, *\*args*,  
*\*\*kwargs*)

Bases: `object`

Client for the Ironic v1 API.

### Parameters

- **endpoint\_override** (*string*) A user-supplied endpoint URL for the ironic service.
- **session** A keystoneauth Session object (must be provided as a keyword argument).

**property** `current_api_version`

Return the current API version in use.

This returns the version of the REST API that the API client is presently set to request. This value may change as a result of API version negotiation.

**property** `is_api_version_negotiated`

Returns True if microversion negotiation has occurred.

**negotiate\_api\_version** ()

Triggers negotiation with the remote API endpoint.

**Returns** the negotiated API version.

## ironicclient.v1.conductor module

**class** `ironicclient.v1.conductor.Conductor` (*manager, info, loaded=False*)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.conductor.ConductorManager` (*api*)

Bases: `ironicclient.common.base.Manager`

**get** (*hostname, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

**list** (*marker=None, limit=None, sort\_key=None, sort\_dir=None, fields=None, detail=False, os\_ironic\_api\_version=None, global\_request\_id=None*)

Retrieve a list of conductors.

### Parameters

- **marker** Optional, the hostname of a conductor, eg the last conductor from a previous result set. Return the next result set.

- **limit**

**The maximum number of results to return per request, if:**

- 1) `limit > 0`, the maximum number of ports to return.
- 2) `limit == 0`, return the entire list of ports.

3) `limit` param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **detail** Optional, boolean whether to return detailed information about conductors.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of conductors.

**resource\_class**  
alias of *Conductor*

## ironicclient.v1.create\_resources module

`ironicclient.v1.create_resources.create_chassis` (*client, chassis\_list*)  
Create chassis from dictionaries.

### Parameters

- **client** ironic client instance.
- **chassis\_list** list of dictionaries to be POSTed to `/chassis` endpoint, if some of them contain `nodes` key, its content is POSTed separately to `/nodes` endpoint.

**Returns** array of exceptions encountered during creation.

`ironicclient.v1.create_resources.create_nodes` (*client, node\_list, chassis\_uuid=None*)

Create nodes from dictionaries.

### Parameters

- **client** ironic client instance.
- **node\_list** list of dictionaries to be POSTed to `/nodes` endpoint, if some of them contain `ports` key, its content is POSTed separately to `/ports` endpoint.
- **chassis\_uuid** UUID of a chassis the nodes should be associated with.

**Returns** array of exceptions encountered during creation.

`ironicclient.v1.create_resources.create_portgroups` (*client, port\_group\_list, node\_uuid*)

Create port groups from dictionaries.

### Parameters

- **client** ironic client instance.
- **portgroup\_list** list of dictionaries to be POSTed to /portgroups endpoint, if some of them contain ports key, its content is POSTed separately to /ports endpoint.
- **node\_uuid** UUID of a node the port groups should be associated with.

**Returns** array of exceptions encountered during creation.

```
ironicclient.v1.create_resources.create_ports (client, port_list, node_uuid,  
                                              portgroup_uuid=None)
```

Create ports from dictionaries.

### Parameters

- **client** ironic client instance.
- **port\_list** list of dictionaries to be POSTed to /ports endpoint.
- **node\_uuid** UUID of a node the ports should be associated with.
- **portgroup\_uuid** UUID of a port group the ports should be associated with, if they are its members.

**Returns** array of exceptions encountered during creation.

```
ironicclient.v1.create_resources.create_resources (client, filenames)
```

Create resources using their JSON or YAML descriptions.

### Parameters

- **client** an instance of ironic client;
- **filenames** a list of filenames containing JSON or YAML resources definitions.

**Raises** ClientException if any operation during files processing/resource creation fails.

```
ironicclient.v1.create_resources.create_single_chassis (client,  
                                                       **params)
```

Call the client to create a chassis.

### Parameters

- **client** ironic client instance.
- **params** dictionary to be POSTed to /chassis endpoint, excluding nodes key.

**Returns** UUID of the created chassis or None in case of exception, and an exception, if it appears.

**Raises** InvalidAttribute, if some parameters passed to clients create\_method are invalid.

**Raises** ClientException, if the creation of the chassis fails.

```
ironicclient.v1.create_resources.create_single_handler (resource_type)
```

Catch errors of the creation of a single resource.



This decorator appends an error (which is an instance of some client exception class) to the return value of the `create_method`, changing the return value from just UUID to (UUID, error), and does some exception handling.

**Parameters** `resource_type` string value, the type of the resource being created, e.g. node, used purely for exception messages.

```
ironicclient.v1.create_resources.create_single_node(client, **params)
```

Call the client to create a node.

**Parameters**

- **client** ironic client instance.
- **params** dictionary to be POSTed to /nodes endpoint, excluding ports and portgroups keys.

**Returns** UUID of the created node or None in case of exception, and an exception, if it appears.

**Raises** InvalidAttribute, if some parameters passed to clients `create_method` are invalid.

**Raises** ClientException, if the creation of the node fails.

```
ironicclient.v1.create_resources.create_single_port(client, **params)
```

Call the client to create a port.

**Parameters**

- **client** ironic client instance.
- **params** dictionary to be POSTed to /ports endpoint.

**Returns** UUID of the created port or None in case of exception, and an exception, if it appears.

**Raises** InvalidAttribute, if some parameters passed to clients `create_method` are invalid.

**Raises** ClientException, if the creation of the port fails.

```
ironicclient.v1.create_resources.create_single_portgroup(client,
                                                         **params)
```

Call the client to create a port group.

**Parameters**

- **client** ironic client instance.
- **params** dictionary to be POSTed to /portgroups endpoint, excluding ports key.

**Returns** UUID of the created port group or None in case of exception, and an exception, if it appears.

**Raises** InvalidAttribute, if some parameters passed to clients `create_method` are invalid.

**Raises** ClientException, if the creation of the portgroup fails.

```
ironicclient.v1.create_resources.load_from_file(filename)
```

Deserialize JSON or YAML from file.

**Parameters** `filename` name of the file containing JSON or YAML.

**Returns** a dictionary deserialized from JSON or YAML.

**Raises** `ClientException` if the file can not be loaded or if its contents is not a valid JSON or YAML, or if the file extension is not supported.

## ironicclient.v1.deploy\_template module

```
class ironicclient.v1.deploy_template.DeployTemplate(manager, info,
                                                    loaded=False)
```

Bases: `ironicclient.common.base.Resource`

```
class ironicclient.v1.deploy_template.DeployTemplateManager(api)
```

Bases: `ironicclient.common.base.CreateManager`

```
delete(template_id, os_ironic_api_version=None, global_request_id=None)
```

```
get(template_id, fields=None, os_ironic_api_version=None, global_request_id=None)
```

```
list(limit=None, marker=None, sort_key=None, sort_dir=None, detail=False,
      fields=None, os_ironic_api_version=None, global_request_id=None)
Retrieve a list of deploy templates.
```

### Parameters

- **marker** Optional, the UUID of a deploy template, eg the last template from a previous result set. Return the next result set.
- **limit**  
**The maximum number of results to return per request, if:**
  - 1) `limit > 0`, the maximum number of deploy templates to return.
  - 2) `limit == 0`, return the entire list of deploy templates.
  - 3) `limit` param is NOT specified (`None`), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about deploy templates.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of deploy templates.

**resource\_class**

alias of *DeployTemplate*

**update** (*template\_id, patch, os\_ironic\_api\_version=None, global\_request\_id=None*)

## ironicclient.v1.driver module

**class** `ironicclient.v1.driver.Driver` (*manager, info, loaded=False*)

Bases: *ironicclient.common.base.Resource*

**class** `ironicclient.v1.driver.DriverManager` (*api*)

Bases: *ironicclient.common.base.Manager*

**delete** (*driver\_name, os\_ironic\_api\_version=None, global\_request\_id=None*)

**get** (*driver\_name, os\_ironic\_api\_version=None, global\_request\_id=None*)

**get\_vendor\_passthru\_methods** (*driver\_name, os\_ironic\_api\_version=None, global\_request\_id=None*)

**list** (*driver\_type=None, detail=None, os\_ironic\_api\_version=None, global\_request\_id=None*)  
Retrieve a list of drivers.

### Parameters

- **driver\_type** Optional, string to filter the drivers by type. Value should be classic or dynamic.
- **detail** Optional, flag whether to return detailed information about drivers. Default is None means not to send the arg to the server due to older versions of the server cannot handle filtering on detail.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of drivers.

**properties** (*driver\_name, os\_ironic\_api\_version=None, global\_request\_id=None*)

**raid\_logical\_disk\_properties** (*driver\_name, os\_ironic\_api\_version=None, global\_request\_id=None*)

Returns the RAID logical disk properties for the driver.

### Parameters

- **driver\_name** Name of the driver.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A dictionary containing the properties that can be mentioned for RAID logical disks and a textual description for them. It returns an empty dictionary on error.

**resource\_class**

alias of *Driver*

**update** (*driver\_name*, *patch*, *http\_method*='PATCH', *os\_ironic\_api\_version*=None, *global\_request\_id*=None)

**vendor\_passthru** (*driver\_name*, *method*, *args*=None, *http\_method*=None, *os\_ironic\_api\_version*=None, *global\_request\_id*=None)

Issue requests for vendor-specific actions on a given driver.

#### Parameters

- **driver\_name** The name of the driver.
- **method** Name of the vendor method.
- **args** Optional. The arguments to be passed to the method.
- **http\_method** The HTTP method to use on the request. Defaults to POST.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

### ironicclient.v1.events module

**class** `ironicclient.v1.events.Event` (*manager*, *info*, *loaded*=False)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.events.EventManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**resource\_class**

alias of *Event*

### ironicclient.v1.node module

**class** `ironicclient.v1.node.Node` (*manager*, *info*, *loaded*=False)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.node.NodeManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**add\_trait** (*node\_ident*, *trait*, *os\_ironic\_api\_version*=None, *global\_request\_id*=None)

Add a trait to a node.

#### Parameters

- **node\_ident** node UUID or name.
- **trait** trait to add to the node.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.

- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**delete** (*node\_id*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get** (*node\_id*, *fields=None*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get\_bios\_setting** (*node\_ident*, *name*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

Get a BIOS setting from a node.

#### Parameters

- **node\_ident** node UUID or name.
- **name** BIOS setting name to get from the node.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**get\_boot\_device** (*node\_uuid*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get\_by\_instance\_uuid** (*instance\_uuid*, *fields=None*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get\_console** (*node\_uuid*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get\_supported\_boot\_devices** (*node\_uuid*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**get\_traits** (*node\_ident*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

Get traits for a node.

#### Parameters

- **node\_ident** node UUID or name.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**get\_vendor\_passthru\_methods** (*node\_ident*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**inject\_nmi** (*node\_uuid*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**list** (*associated=None*, *maintenance=None*, *marker=None*, *limit=None*, *detail=False*, *sort\_key=None*, *sort\_dir=None*, *fields=None*, *provision\_state=None*, *driver=None*, *resource\_class=None*, *chassis=None*, *fault=None*, *os\_ironic\_api\_version=None*, *conductor\_group=None*, *conductor=None*, *owner=None*, *retired=None*, *lessee=None*, *global\_request\_id=None*)

Retrieve a list of nodes.

#### Parameters

- **associated** Optional. Either a Boolean or a string representation of a Boolean that indicates whether to return a list of associated (True or True) or unassociated (False or False) nodes.
- **maintenance** Optional. Either a Boolean or a string representation of a Boolean that indicates whether to return nodes in maintenance mode (True or True), or not in maintenance mode (False or False).
- **retired** Optional. Either a Boolean or a string representation of a Boolean that indicates whether to return retired nodes (True or True).
- **provision\_state** Optional. String value to get only nodes in that provision state.
- **marker** Optional, the UUID of a node, eg the last node from a previous result set. Return the next result set.
- **limit**

**The maximum number of results to return per** request, if:

- 1) `limit > 0`, the maximum number of nodes to return.
- 2) `limit == 0`, return the entire list of nodes.
- 3) `limit` param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).

- **detail** Optional, boolean whether to return detailed information about nodes.
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **driver** Optional. String value to get only nodes using that driver.
- **resource\_class** Optional. String value to get only nodes with the given resource class set.
- **chassis** Optional, the UUID of a chassis. Used to get only nodes of this chassis.
- **fault** Optional. String value to get only nodes with specified fault.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.
- **conductor\_group** Optional. String value to get only nodes with the given conductor group set.
- **conductor** Optional. String value to get only nodes mapped to the given conductor.

- **owner** Optional. String value to get only nodes mapped to a specific owner.
- **lessee** Optional. String value to get only nodes mapped to a specific lessee.

**Returns** A list of nodes.

**list\_bios\_settings** (*node\_ident*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)  
 List all BIOS settings from a node.

**Parameters**

- **node\_ident** node UUID or name.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**list\_ports** (*node\_id*, *marker=None*, *limit=None*, *sort\_key=None*, *sort\_dir=None*,  
*detail=False*, *fields=None*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)  
 List all the ports for a given node.

**Parameters**

- **node\_id** Name or UUID of the node.
- **marker** Optional, the UUID of a port, eg the last port from a previous result set. Return the next result set.

- **limit**

**The maximum number of results to return per request, if:**

- 1) **limit** > 0, the maximum number of ports to return.
- 2) **limit** == 0, return the entire list of ports.
- 3) **limit** param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see Ironics api.max\_limit option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either asc (the default) or desc.
- **detail** Optional, boolean whether to return detailed information about ports.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of ports.

**list\_volume\_connectors** (*node\_id*, *marker=None*, *limit=None*, *sort\_key=None*,  
*sort\_dir=None*, *detail=False*, *fields=None*,  
*os\_ironic\_api\_version=None*, *global\_request\_id=None*)

List all the volume connectors for a given node.

#### Parameters

- **node\_id** Name or UUID of the node.
- **marker** Optional, the UUID of a volume connector, eg the last volume connector from a previous result set. Return the next result set.
- **limit**  
**The maximum number of results to return per request, if:**
  - 1) **limit** > 0, the maximum number of volume connectors to return.
  - 2) **limit** == 0, return the entire list of volume connectors.
  - 3) **limit** param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see Ironics api.max\_limit option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either asc (the default) or desc.
- **detail** Optional, boolean whether to return detailed information about volume connectors.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of volume connectors.

**list\_volume\_targets** (*node\_id*, *marker=None*, *limit=None*, *sort\_key=None*,  
*sort\_dir=None*, *detail=False*, *fields=None*,  
*os\_ironic\_api\_version=None*, *global\_request\_id=None*)

List all the volume targets for a given node.

#### Parameters

- **node\_id** Name or UUID of the node.
- **marker** Optional, the UUID of a volume target, eg the last volume target from a previous result set. Return the next result set.
- **limit**  
**The maximum number of results to return per request, if:**
  - 1) **limit** > 0, the maximum number of volume targets to return.
  - 2) **limit** == 0, return the entire list of volume targets.



3) `limit` param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about volume targets.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of volume targets.

**remove\_all\_traits** (*node\_ident*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)  
 Remove all traits from a node.

**Parameters**

- **node\_ident** node UUID or name.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**remove\_trait** (*node\_ident*, *trait*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)  
 Remove a trait from a node.

**Parameters**

- **node\_ident** node UUID or name.
- **trait** trait to remove from the node.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**resource\_class**

alias of *Node*

**set\_boot\_device** (*node\_uuid*, *boot\_device*, *persistent=False*,  
*os\_ironic\_api\_version=None*, *global\_request\_id=None*)

**set\_console\_mode** (*node\_uuid*, *enabled*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)

Set the console mode for the node.

**Parameters**

- **node\_uuid** The UUID of the node.
- **enabled** Either a Boolean or a string representation of a Boolean. True to enable the console; False to disable.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**set\_maintenance** (*node\_id*, *state*, *maint\_reason=None*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)

Set the maintenance mode for the node.

#### Parameters

- **node\_id** The UUID of the node.
- **state** the maintenance mode; either a Boolean or a string representation of a Boolean (eg, true, on, false, off). True to put the node in maintenance mode; False to take the node out of maintenance mode.
- **maint\_reason** Optional string. Reason for putting node into maintenance mode.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Raises** InvalidAttribute if state is an invalid string (that doesnt represent a Boolean).

**set\_power\_state** (*node\_id*, *state*, *soft=False*, *timeout=None*,  
*os\_ironic\_api\_version=None*, *global\_request\_id=None*)

Sets power state for a node.

#### Parameters

- **node\_id** Node identifier
- **state** One of target power state, on, off, or reboot
- **soft** The flag for graceful power off or reboot
- **timeout** The timeout (in seconds) positive integer value (> 0)
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Raises** ValueError if soft or timeout option is invalid

**Returns** The status of the request

**set\_provision\_state** (*node\_uuid*, *state*, *configdrive=None*, *cleansteps=None*,  
*rescue\_password=None*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)

Set the provision state for the node.

### Parameters

- **node\_uuid** The UUID or name of the node.
- **state** The desired provision state. One of active, deleted, rebuild, inspect, provide, manage, clean, abort, rescue, unrescue.
- **configdrive** A gzipped, base64-encoded configuration drive string OR the path to the configuration drive file OR the path to a directory containing the config drive files OR a dictionary to build config drive from. In case its a directory, a config drive will be generated from it. In case its a dictionary, a config drive will be generated on the server side (requires API version 1.56). This is only valid when setting state to active.
- **cleansteps** The clean steps as a list of clean-step dictionaries; each dictionary should have keys interface and step, and optional key args. This must be specified (and is only valid) when setting provision-state to clean.
- **rescue\_password** A string to be used as the login password inside the rescue ramdisk once a node is rescued. This must be specified (and is only valid) when setting state to rescue.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Raises** InvalidAttribute if there was an error with the clean steps

**Returns** The status of the request

**set\_target\_raid\_config**(*node\_ident*, *target\_raid\_config*,  
*os\_ironic\_api\_version=None*, *global\_request\_id=None*)  
 Sets target\_raid\_config for a node.

### Parameters

- **node\_ident** Node identifier
- **target\_raid\_config** A dictionary with the target RAID configuration; may be empty.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** status of the request

**set\_traits**(*node\_ident*, *traits*, *os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)  
 Set traits for a node.

Removes any existing traits and adds the traits passed in to this method.

### Parameters

- **node\_ident** node UUID or name.
- **traits** list of traits to add to the node.

- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**states** (*node\_uuid, os\_ironic\_api\_version=None, global\_request\_id=None*)

**update** (*node\_id, patch, http\_method='PATCH', os\_ironic\_api\_version=None, reset\_interfaces=None, global\_request\_id=None*)

**validate** (*node\_uuid, os\_ironic\_api\_version=None, global\_request\_id=None*)

**vendor\_passthru** (*node\_id, method, args=None, http\_method=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Issue requests for vendor-specific actions on a given node.

#### Parameters

- **node\_id** The UUID of the node.
- **method** Name of the vendor method.
- **args** Optional. The arguments to be passed to the method.
- **http\_method** The HTTP method to use on the request. Defaults to POST.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**vif\_attach** (*node\_ident, vif\_id, os\_ironic\_api\_version=None, global\_request\_id=None, \*\*kwargs*)

Attach VIF to a given node.

#### Parameters

- **node\_ident** The UUID or Name of the node.
- **vif\_id** The UUID or Name of the VIF to attach.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.
- **kwargs** A dictionary containing the attributes of the resource that will be created.

**vif\_detach** (*node\_ident, vif\_id, os\_ironic\_api\_version=None, global\_request\_id=None*)

Detach VIF from a given node.

#### Parameters

- **node\_ident** The UUID or Name of the node.
- **vif\_id** The UUID or Name of the VIF to detach.

- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**vif\_list** (*node\_ident*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

List VIFs attached to a given node.

#### Parameters

- **node\_ident** The UUID or Name of the node.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**wait\_for\_provision\_state** (*node\_ident*, *expected\_state*, *timeout=0*,  
*poll\_interval=2*, *poll\_delay\_function=None*,  
*fail\_on\_unexpected\_state=True*,  
*os\_ironic\_api\_version=None*,  
*global\_request\_id=None*)

Helper function to wait for a node to reach a given state.

Polls Ironic API in a loop until node gets to a requested state.

Fails in the following cases: \* Timeout (if provided) is reached \* Nodes last\_error gets set to a non-empty value \* Unexpected stable state is reached and fail\_on\_unexpected\_state is on \* Error state is reached (if its not equal to expected\_state)

#### Parameters

- **node\_ident** node UUID or name
- **expected\_state** expected final provision state
- **timeout** timeout in seconds, no timeout if 0
- **poll\_interval** interval in seconds between 2 poll
- **poll\_delay\_function** function to use to wait between polls (defaults to time.sleep). Should take one argument - delay time in seconds. Any exceptions raised inside it will abort the wait.
- **fail\_on\_unexpected\_state** whether to fail if the nodes reaches a different stable state.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Raises** StateTransitionFailed if node reached an error state

**Raises** StateTransitionTimeout on timeout

## ironicclient.v1.port module

**class** `ironicclient.v1.port.Port` (*manager, info, loaded=False*)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.port.PortManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**delete** (*port\_id, os\_ironic\_api\_version=None, global\_request\_id=None*)

**get** (*port\_id, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

**get\_by\_address** (*address, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

**list** (*address=None, limit=None, marker=None, sort\_key=None, sort\_dir=None, detail=False, fields=None, node=None, portgroup=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Retrieve a list of ports.

### Parameters

- **address** Optional, MAC address of a port, to get the port which has this MAC address
- **marker** Optional, the UUID of a port, eg the last port from a previous result set. Return the next result set.
- **limit**  
**The maximum number of results to return per request, if:**
  - 1) `limit > 0`, the maximum number of ports to return.
  - 2) `limit == 0`, return the entire list of ports.
  - 3) `limit` param is NOT specified (`None`), the number of items returned respect the maximum imposed by the Ironic API (see Ironics `api.max_limit` option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about ports.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **node** Optional, name or UUID of a node. Used to get ports of this node.
- **portgroup** Optional, name or UUID of a portgroup. Used to get ports of this portgroup.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of ports.

**resource\_class**

alias of *Port*

**update** (*port\_id, patch, os\_ironic\_api\_version=None, global\_request\_id=None*)

## ironicclient.v1.portgroup module

**class** `ironicclient.v1.portgroup.Portgroup` (*manager, info, loaded=False*)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.portgroup.PortgroupManager` (*api*)

Bases: `ironicclient.common.base.CreateManager`

**delete** (*portgroup\_id, os\_ironic\_api\_version=None, global\_request\_id=None*)

Delete the Portgroup from the DB.

### Parameters

- **portgroup\_id** The UUID or name of a portgroup.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**get** (*portgroup\_id, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Get a port group with the specified identifier.

### Parameters

- **portgroup\_id** The UUID or name of a portgroup.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** a *Portgroup* object.

**get\_by\_address** (*address, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)

Get a port group with the specified MAC address.

### Parameters

- **address** The MAC address of a portgroup.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.

- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** a *Portgroup* object.

**list** (*node=None, address=None, limit=None, marker=None, sort\_key=None, sort\_dir=None, detail=False, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)  
Retrieve a list of portgroups.

#### Parameters

- **node** Optional, UUID or name of a node, to get the portgroups for that node.
- **address** Optional, MAC address of a portgroup, to get the portgroup which has this MAC address.
- **marker** Optional, the UUID of a portgroup, eg the last portgroup from a previous result set. Return the next result set.
- **limit**

**The maximum number of results to return per** request, if:

- 1) **limit** > 0, the maximum number of portgroups to return.
- 2) **limit** == 0, return the entire list of portgroups.
- 3) **limit** == None, the number of items returned respect the maximum imposed by the Ironic API (see Ironics `api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either asc (the default) or desc.
- **detail** Optional, boolean whether to return detailed information about portgroups.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of portgroups.

**Raises** `InvalidAttribute` if a subset of fields is requested with detail option set.

**list\_ports** (*portgroup\_id, marker=None, limit=None, sort\_key=None, sort\_dir=None, detail=False, fields=None, os\_ironic\_api\_version=None, global\_request\_id=None*)  
List all the ports for a given portgroup.

#### Parameters

- **portgroup\_id** Name or UUID of the portgroup.



- **marker** Optional, the UUID of a port, eg the last port from a previous result set. Return the next result set.
- **limit**

The maximum number of results to return per request, if:

  - 1) limit > 0, the maximum number of ports to return.
  - 2) limit == 0, return the entire list of ports.
  - 3) limit param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see Ironics api.max\_limit option).
- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either asc (the default) or desc.
- **detail** Optional, boolean whether to return detailed information about ports.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

**Returns** A list of ports.

**resource\_class**

alias of *Portgroup*

**update** (*portgroup\_id*, *patch*, *os\_ironic\_api\_version=None*, *global\_request\_id=None*)

Update the Portgroup.

**Parameters**

- **portgroup\_id** The UUID or name of a portgroup.
- **patch** The patch request with updates.
- **os\_ironic\_api\_version** String version (e.g. 1.35) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form req-<UUID>) to use for the request.

## ironicclient.v1.resource\_fields module

```
class ironicclient.v1.resource_fields.Resource (field_ids,  
                                              sort_excluded=None,  
                                              override_labels=None)
```

Bases: object

Resource class

This class is used to manage the various fields that a resource (e.g. Chassis, Node, Port) contains. An individual field consists of a field\_id (key) and a label (value). The caller only provides the field\_ids when instantiating the object.

Ordering of the field\_ids will be preserved as specified by the caller.

It also provides the ability to exclude some of these fields when they are being used for sorting.

```
FIELDS = {'address': 'Address', 'alive': 'Alive', 'allocation_uuid': 'All
```

```
property fields
```

```
property labels
```

```
property sort_fields
```

```
property sort_labels
```

## ironicclient.v1.utils module

### ironicclient.v1.volume\_connector module

```
class ironicclient.v1.volume_connector.VolumeConnector (manager, info,  
                                                         loaded=False)
```

Bases: *ironicclient.common.base.Resource*

```
class ironicclient.v1.volume_connector.VolumeConnectorManager (api)  
Bases: ironicclient.common.base.CreateManager
```

```
delete (volume_connector_id, os_ironic_api_version=None, global_request_id=None)
```

```
get (volume_connector_id, fields=None, os_ironic_api_version=None,  
      global_request_id=None)
```

```
list (node=None, limit=None, marker=None, sort_key=None, sort_dir=None, de-  
      tail=False, fields=None, os_ironic_api_version=None, global_request_id=None)  
Retrieve a list of volume connector.
```

#### Parameters

- **node** Optional, UUID or name of a node, to get volume connectors for this node only.
- **marker** Optional, the UUID of a volume connector, eg the last volume connector from a previous result set. Return the next result set.
- **limit**

**The maximum number of results to return per request, if:**

- 1) limit > 0, the maximum number of volume connectors to return.

- 2) `limit == 0`, return the entire list of volume connectors.
- 3) `limit` param is NOT specified (None), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about volume connectors.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of volume connectors.

**resource\_class**

alias of `VolumeConnector`

**update** (`volume_connector_id`, `patch`, `os_ironic_api_version=None`,  
`global_request_id=None`)

## ironicclient.v1.volume\_target module

**class** `ironicclient.v1.volume_target.VolumeTarget` (`manager`, `info`,  
`loaded=False`)

Bases: `ironicclient.common.base.Resource`

**class** `ironicclient.v1.volume_target.VolumeTargetManager` (`api`)

Bases: `ironicclient.common.base.CreateManager`

**delete** (`volume_target_id`, `os_ironic_api_version=None`, `global_request_id=None`)

**get** (`volume_target_id`, `fields=None`, `os_ironic_api_version=None`,  
`global_request_id=None`)

**list** (`node=None`, `limit=None`, `marker=None`, `sort_key=None`, `sort_dir=None`, `de-  
tail=False`, `fields=None`, `os_ironic_api_version=None`, `global_request_id=None`)

Retrieve a list of volume target.

### Parameters

- **node** Optional, UUID or name of a node, to get volume targets for this node only.
- **marker** Optional, the UUID of a volume target, eg the last volume target from a previous result set. Return the next result set.
- **limit**

The maximum number of results to return per request, if:

- 1) `limit > 0`, the maximum number of volume targets to return.
- 2) `limit == 0`, return the entire list of volume targets.
- 3) `limit` param is NOT specified (`None`), the number of items returned respect the maximum imposed by the Ironic API (see `Ironics api.max_limit` option).

- **sort\_key** Optional, field used for sorting.
- **sort\_dir** Optional, direction of sorting, either `asc` (the default) or `desc`.
- **detail** Optional, boolean whether to return detailed information about volume targets.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when `detail` is set.
- **os\_ironic\_api\_version** String version (e.g. `1.35`) to use for the request. If not specified, the clients default is used.
- **global\_request\_id** String containing global request ID header value (in form `req-<UUID>`) to use for the request.

**Returns** A list of volume targets.

**resource\_class**

alias of `VolumeTarget`

**update** (`volume_target_id`, `patch`, `os_ironic_api_version=None`,  
`global_request_id=None`)

## Module contents

### Submodules

#### `ironicclient.client` module

`ironicclient.client.Client` (`version`, `endpoint_override=None`, `session=None`, `*args`,  
`**kwargs`)

Create a client of an appropriate version.

This call requires a session. If you want it to be created, use `get_client` instead.

#### Parameters

- **endpoint\_override** A bare metal endpoint to use.
- **session** A keystoneauth session to use. This argument is actually required and is marked optional only for backward compatibility.
- **args** Other arguments to pass to the HTTP client. Not recommended, use `kwargs` instead.
- **kwargs** Other keyword arguments to pass to the HTTP client (e.g. `insecure`).

```
ironicclient.client.get_client (api_version, auth_type=None,
                                os_ironic_api_version=None, max_retries=None,
                                retry_interval=None, session=None,
                                valid_interfaces=None, interface=None, ser-
                                vice_type=None, region_name=None, addi-
                                tional_headers=None, global_request_id=None,
                                **kwargs)
```

Get an authenticated client, based on the credentials.

#### Parameters

- **api\_version** the API version to use. Valid value: 1.
- **auth\_type** type of keystoneauth auth plugin loader to use.
- **os\_ironic\_api\_version** ironic API version to use.
- **max\_retries** Maximum number of retries in case of conflict error
- **retry\_interval** Amount of time (in seconds) between retries in case of conflict error.
- **session** An existing keystoneauth session. Will be created from kwargs if not provided.
- **valid\_interfaces** List of valid endpoint interfaces to use if the bare metal endpoint is not provided.
- **interface** An alias for valid\_interfaces.
- **service\_type** Bare metal endpoint service type.
- **region\_name** Name of the region to use when searching the bare metal endpoint.
- **additional\_headers** Additional headers that should be attached to every request passing through the client. Headers of the same name specified per request will take priority.
- **global\_request\_id** A header (in the form of req- $\$$ uuid) that will be passed on all requests. Enables cross project request id tracking.
- **kwargs** all the other params that are passed to keystoneauth for session construction.

#### ironicclient.exc module

```
ironicclient.exc.AmbiguousAuthSystem
```

alias of `ironicclient.exc.AmbiguousAuthSystem`

```
exception ironicclient.exc.AmbiguousAuthSystem
```

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Could not obtain token and endpoint using provided credentials.

```
exception ironicclient.exc.InvalidAttribute
```

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

**exception** `ironicclient.exc.StateTransitionFailed`

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Failed to reach a requested provision state.

**exception** `ironicclient.exc.StateTransitionTimeout`

Bases: `ironicclient.common.apiclient.exceptions.ClientException`

Timed out while waiting for a requested provision state.

`ironicclient.exc.from_response` (*response*, *message=None*, *traceback=None*,  
*method=None*, *url=None*)

Return an `HttpError` instance based on response from `httplib/requests`.

## ironicclient.shell module

**class** `ironicclient.shell.App`

Bases: `cliff.app.App`

**build\_option\_parser** (*description*, *version*, *argparse\_kwargs=None*)

Return an `argparse` option parser for this application.

Subclasses may override this method to extend the parser with more global options.

### Parameters

- **description** (*str*) full description of the application
- **version** (*str*) version number for the application
- **argparse\_kwargs** extra keyword argument passed to the `ArgumentParser` constructor

**initialize\_app** (*argv*)

Hook for subclasses to take global initialization action after the arguments are parsed but before a command is run. Invoked only once, even in interactive mode.

**Parameters** *argv* List of arguments, including the subcommand to run. Empty for interactive mode.

**class** `ironicclient.shell.ClientManager` (*cloud\_region*, *options*)

Bases: `object`

**property** `baremetal`

**property** `baremetal_introspection`

**class** `ironicclient.shell.CommandManager` (*namespace*, *vert\_underscores=True*) *con-*

Bases: `cliff.commandmanager.CommandManager`

**load\_commands** (*namespace*)

Load all the commands from an endpoint

`ironicclient.shell.main` (*argv=['-W', '-b', 'latex', 'doc/source', 'doc/build/pdf']*)

## Module contents





## INDICES AND TABLES

- genindex
- modindex
- search



## PYTHON MODULE INDEX

### C

`ironicclient.client`, 144  
`ironicclient.common`, 93  
`ironicclient.common.apiclient`, 88  
`ironicclient.common.apiclient.base`,  
74  
`ironicclient.common.apiclient.exceptions`,  
76  
`ironicclient.common.base`, 88  
`ironicclient.common.filecache`, 89  
`ironicclient.common.http`, 89  
`ironicclient.common.i18n`, 90  
`ironicclient.common.utils`, 90

### e

`ironicclient.exc`, 145

### i

`ironicclient`, 147

### O

`ironicclient.osc`, 117  
`ironicclient.osc.plugin`, 117  
`ironicclient.osc.v1`, 117  
`ironicclient.osc.v1.baremetal_allocation`,  
93  
`ironicclient.osc.v1.baremetal_chassis`,  
95  
`ironicclient.osc.v1.baremetal_conductor`,  
96  
`ironicclient.osc.v1.baremetal_create`,  
97  
`ironicclient.osc.v1.baremetal_deploy_template`,  
97  
`ironicclient.osc.v1.baremetal_driver`,  
99  
`ironicclient.osc.v1.baremetal_node`,  
100  
`ironicclient.osc.v1.baremetal_port`,  
110

`ironicclient.osc.v1.baremetal_portgroup`,  
112  
`ironicclient.osc.v1.baremetal_volume_connector`,  
114  
`ironicclient.osc.v1.baremetal_volume_target`,  
115

### S

`ironicclient.shell`, 146

### V

`ironicclient.v1`, 144  
`ironicclient.v1.allocation`, 117  
`ironicclient.v1.chassis`, 120  
`ironicclient.v1.client`, 122  
`ironicclient.v1.conductor`, 122  
`ironicclient.v1.create_resources`,  
123  
`ironicclient.v1.deploy_template`,  
126  
`ironicclient.v1.driver`, 127  
`ironicclient.v1.events`, 128  
`ironicclient.v1.node`, 128  
`ironicclient.v1.port`, 138  
`ironicclient.v1.portgroup`, 139  
`ironicclient.v1.resource_fields`,  
142  
`ironicclient.v1.utils`, 142  
`ironicclient.v1.volume_connector`,  
142  
`ironicclient.v1.volume_target`, 143



## Symbols

- address
  - openstack-baremetal-port-group-show command line option, 55
  - openstack-baremetal-port-group-unset command line option, 55
  - openstack-baremetal-port-show command line option, 59
- address <address>
  - openstack-baremetal-port-set command line option, 57
- address <mac-address>
  - openstack-baremetal-port-group-create command line option, 52
  - openstack-baremetal-port-group-list command line option, 53
  - openstack-baremetal-port-group-set command line option, 54
  - openstack-baremetal-port-list command line option, 57
- all
  - openstack-baremetal-node-remove-trait command line option, 38
- arg <key=value>
  - openstack-baremetal-driver-passthru-cap command line option, 18
  - openstack-baremetal-node-passthru-cap command line option, 35
- associated
  - openstack-baremetal-node-list command line option, 33
- automated-clean
  - openstack-baremetal-node-create command line option, 30
  - openstack-baremetal-node-set command line option, 41
  - openstack-baremetal-node-unset command line option, 46
- bios-interface
  - openstack-baremetal-node-unset command line option, 45
- bios-interface <bios\_interface>
  - openstack-baremetal-node-create command line option, 29
  - openstack-baremetal-node-set command line option, 39
- boot-index <boot index>
  - openstack-baremetal-volume-target-create command line option, 65
  - openstack-baremetal-volume-target-set command line option, 67
- boot-interface
  - openstack-baremetal-node-unset command line option, 45
- boot-interface <boot\_interface>
  - openstack-baremetal-node-create command line option, 29
  - openstack-baremetal-node-set command line option, 39
- candidate-node
  - <CANDIDATE\_NODES>
  - openstack-baremetal-allocation-create command line option, 9
- chassis <chassis UUID>
  - openstack-baremetal-node-list command line option, 33
- chassis-uuid
  - openstack-baremetal-node-unset command line option, 45
- chassis-uuid <chassis UUID>
  - openstack-baremetal-node-set command line option, 39
- chassis-uuid <chassis>
  - openstack-baremetal-node-create command line option, 29
- clean-steps <clean-steps>
  - openstack-baremetal-node-clean command line option, 26
- column COLUMN
  - openstack-baremetal-allocation-create command line option, 8
  - openstack-baremetal-allocation-list

command line option, 10  
 openstack-baremetal-allocation-show  
     command line option, 11  
 openstack-baremetal-chassis-create  
     command line option, 13  
 openstack-baremetal-chassis-list  
     command line option, 14  
 openstack-baremetal-chassis-show  
     command line option, 15  
 openstack-baremetal-driver-list  
     command line option, 17  
 openstack-baremetal-driver-passthru-cabm  
     command line option, 18  
 openstack-baremetal-driver-passthru-list  
     command line option, 19  
 openstack-baremetal-driver-property-list  
     command line option, 20  
 openstack-baremetal-driver-raid-property-list  
     command line option, 21  
 openstack-baremetal-driver-show  
     command line option, 22  
 openstack-baremetal-node-bios-setting-bios  
     command line option, 23  
 openstack-baremetal-node-bios-setting-show  
     command line option, 24  
 openstack-baremetal-node-boot-device-show  
     command line option, 25  
 openstack-baremetal-node-console-show  
     command line option, 27  
 openstack-baremetal-node-create  
     command line option, 28  
 openstack-baremetal-node-list  
     command line option, 32  
 openstack-baremetal-node-passthru-list  
     command line option, 35  
 openstack-baremetal-node-show  
     command line option, 42  
 openstack-baremetal-node-trait-list  
     command line option, 43  
 openstack-baremetal-node-validate  
     command line option, 47  
 openstack-baremetal-node-vif-list  
     command line option, 49  
 openstack-baremetal-port-create  
     command line option, 50  
 openstack-baremetal-port-group-create  
     command line option, 51  
 openstack-baremetal-port-group-list  
     command line option, 53  
 openstack-baremetal-port-group-show  
     command line option, 55  
 openstack-baremetal-port-list  
     command line option, 56  
 openstack-baremetal-port-show  
     command line option, 58  
 openstack-baremetal-volume-connector-create  
     command line option, 60  
 openstack-baremetal-volume-connector-list  
     command line option, 61  
 openstack-baremetal-volume-connector-show  
     command line option, 63  
 openstack-baremetal-volume-target-create  
     command line option, 64  
 openstack-baremetal-volume-target-list  
     command line option, 65  
 openstack-baremetal-volume-target-show  
     command line option, 67  
 --conductor <conductor>  
 openstack-baremetal-node-list  
     command line option, 33  
 --conductor-group  
 openstack-baremetal-node-unset  
     command line option, 46  
 --conductor-group  
 <conductor\_group>  
 openstack-baremetal-node-create  
     command line option, 30  
 openstack-baremetal-node-list  
     command line option, 33  
 openstack-baremetal-node-set  
     command line option, 41  
 --config-drive <config-drive>  
 openstack-baremetal-node-deploy  
     command line option, 31  
 openstack-baremetal-node-rebuild  
     command line option, 37  
 --connector-id <connector id>  
 openstack-baremetal-volume-connector-create  
     command line option, 60  
 openstack-baremetal-volume-connector-set  
     command line option, 62  
 --console-interface  
 openstack-baremetal-node-unset  
     command line option, 45  
 <console\_interface>  
 openstack-baremetal-node-create  
     command line option, 29  
 openstack-baremetal-node-set  
     command line option, 40  
 deploy-interface  
 openstack-baremetal-node-unset

```

    command line option, 45
--deploy-interface
    <deploy_interface>
    openstack-baremetal-node-create
        command line option, 29
    openstack-baremetal-node-set
        command line option, 40
--description
    openstack-baremetal-chassis-unset
        command line option, 16
    openstack-baremetal-node-unset
        command line option, 46
--description <description>
    openstack-baremetal-chassis-create
        command line option, 13
    openstack-baremetal-chassis-set
        command line option, 15
    openstack-baremetal-node-create
        command line option, 30
    openstack-baremetal-node-set
        command line option, 41
--description-contains
    <description_contains>
    openstack-baremetal-node-list
        command line option, 33
--driver <driver>
    openstack-baremetal-node-create
        command line option, 29
    openstack-baremetal-node-list
        command line option, 33
    openstack-baremetal-node-set
        command line option, 39
--driver-info <key=value>
    openstack-baremetal-node-create
        command line option, 29
    openstack-baremetal-node-set
        command line option, 41
--driver-info <key>
    openstack-baremetal-node-unset
        command line option, 45
--extra <key=value>
    openstack-baremetal-allocation-create
        command line option, 9
    openstack-baremetal-allocation-set
        command line option, 11
    openstack-baremetal-chassis-create
        command line option, 13
    openstack-baremetal-chassis-set
        command line option, 15
    openstack-baremetal-node-create
        command line option, 29
    openstack-baremetal-node-set
        command line option, 41
    openstack-baremetal-port-create
        command line option, 50
    openstack-baremetal-port-group-create
        command line option, 52
    openstack-baremetal-port-group-set
        command line option, 54
    openstack-baremetal-port-set
        command line option, 57
    openstack-baremetal-volume-connector-create
        command line option, 60
    openstack-baremetal-volume-connector-set
        command line option, 62
    openstack-baremetal-volume-target-create
        command line option, 65
    openstack-baremetal-volume-target-set
        command line option, 67
--extra <key>
    openstack-baremetal-allocation-unset
        command line option, 12
    openstack-baremetal-chassis-unset
        command line option, 16
    openstack-baremetal-node-unset
        command line option, 45
    openstack-baremetal-port-group-unset
        command line option, 56
    openstack-baremetal-port-unset
        command line option, 59
    openstack-baremetal-volume-connector-unset
        command line option, 63
    openstack-baremetal-volume-target-unset
        command line option, 68
--fault <fault>
    openstack-baremetal-node-list
        command line option, 33
--fields <field>
    openstack-baremetal-allocation-list
        command line option, 11
    openstack-baremetal-allocation-show
        command line option, 12
    openstack-baremetal-chassis-list
        command line option, 14
    openstack-baremetal-chassis-show
        command line option, 16
    openstack-baremetal-node-list
        command line option, 34
    openstack-baremetal-node-show
        command line option, 42
    openstack-baremetal-port-group-list
        command line option, 53

```





openstack-baremetal-driver-passthru-list command line option, 65  
 command line option, 19 openstack-baremetal-volume-target-show  
 openstack-baremetal-driver-property-list command line option, 67  
 command line option, 20 --http-method <http-method>  
 openstack-baremetal-driver-raid-property-list openstack-baremetal-driver-passthru-call  
 command line option, 21 command line option, 18  
 openstack-baremetal-driver-show openstack-baremetal-node-passthru-call  
 command line option, 22 command line option, 35  
 openstack-baremetal-node-bios-settings openstack-baremetal-node-unset  
 command line option, 23 inspect\_interface  
 openstack-baremetal-node-bios-setting-show command line option, 45  
 command line option, 24 --inspect-interface  
 openstack-baremetal-node-boot-device-show inspect\_interface>  
 command line option, 25 openstack-baremetal-node-create  
 openstack-baremetal-node-console-show command line option, 29  
 command line option, 27 openstack-baremetal-node-set  
 openstack-baremetal-node-create command line option, 40  
 command line option, 28 --instance  
 openstack-baremetal-node-list openstack-baremetal-node-show  
 command line option, 32 command line option, 42  
 openstack-baremetal-node-passthru-list instance-info <key=value>  
 command line option, 35 openstack-baremetal-node-set  
 openstack-baremetal-node-show command line option, 41  
 command line option, 42 --instance-info <key>  
 openstack-baremetal-node-trait-list openstack-baremetal-node-unset  
 command line option, 43 command line option, 45  
 openstack-baremetal-node-validate-instance-uuid  
 command line option, 47 openstack-baremetal-node-unset  
 openstack-baremetal-node-vif-list command line option, 45  
 command line option, 48 --instance-uuid <uuid>  
 openstack-baremetal-port-create openstack-baremetal-node-set  
 command line option, 49 command line option, 39  
 openstack-baremetal-port-group-create smartnic  
 command line option, 51 openstack-baremetal-port-create  
 openstack-baremetal-port-group-list command line option, 50  
 command line option, 53 openstack-baremetal-port-set  
 openstack-baremetal-port-group-show command line option, 58  
 command line option, 55 openstack-baremetal-port-unset  
 openstack-baremetal-port-list command line option, 59  
 command line option, 56 --lessee  
 openstack-baremetal-port-show openstack-baremetal-node-unset  
 command line option, 58 command line option, 46  
 openstack-baremetal-volume-connector-create create\_lessee>  
 command line option, 60 openstack-baremetal-node-create  
 openstack-baremetal-volume-connector-list command line option, 30  
 command line option, 61 openstack-baremetal-node-list  
 openstack-baremetal-volume-connector-show command line option, 33  
 command line option, 63 openstack-baremetal-node-set  
 openstack-baremetal-volume-target-create command line option, 41  
 command line option, 64 --limit <limit>  
 openstack-baremetal-volume-target-list openstack-baremetal-allocation-list

```

    command line option, 10
openstack-baremetal-chassis-list
    command line option, 14
openstack-baremetal-node-list
    command line option, 32
openstack-baremetal-port-group-list
    command line option, 53
openstack-baremetal-port-list
    command line option, 57
openstack-baremetal-volume-connector-list
    command line option, 62
openstack-baremetal-volume-target-list
    command line option, 66
--local-link-connection
    <key=value>
openstack-baremetal-port-create
    command line option, 50
openstack-baremetal-port-set
    command line option, 58
--long
openstack-baremetal-allocation-list
    command line option, 10
openstack-baremetal-chassis-list
    command line option, 14
openstack-baremetal-driver-list
    command line option, 17
openstack-baremetal-node-list
    command line option, 33
openstack-baremetal-port-group-list
    command line option, 53
openstack-baremetal-port-list
    command line option, 57
openstack-baremetal-volume-connector-list
    command line option, 62
openstack-baremetal-volume-target-list
    command line option, 66
--maintenance
    openstack-baremetal-node-list
        command line option, 33
--management-interface
    openstack-baremetal-node-unset
        command line option, 46
--management-interface
    <management_interface>
openstack-baremetal-node-create
    command line option, 29
openstack-baremetal-node-set
    command line option, 40
--marker <allocation>
    openstack-baremetal-allocation-list
        command line option, 10
    --marker <chassis>
openstack-baremetal-chassis-list
    command line option, 14
    --marker <node>
openstack-baremetal-node-list
    command line option, 33
    --marker <port group>
openstack-baremetal-port-group-list
    command line option, 53
    --marker <port>
openstack-baremetal-port-list
    command line option, 57
    --marker <volume connector>
openstack-baremetal-volume-connector-list
    command line option, 62
    --marker <volume target>
openstack-baremetal-volume-target-list
    command line option, 66
--max-width <integer>
openstack-baremetal-allocation-create
    command line option, 8
openstack-baremetal-allocation-list
    command line option, 10
openstack-baremetal-allocation-show
    command line option, 11
openstack-baremetal-chassis-create
    command line option, 13
openstack-baremetal-chassis-list
    command line option, 14
openstack-baremetal-chassis-show
    command line option, 16
openstack-baremetal-driver-list
    command line option, 17
openstack-baremetal-driver-passthru-call
    command line option, 18
openstack-baremetal-driver-passthru-list
    command line option, 19
openstack-baremetal-driver-property-list
    command line option, 20
openstack-baremetal-driver-raid-property-list
    command line option, 21
openstack-baremetal-driver-show
    command line option, 22
openstack-baremetal-node-bios-setting-list
    command line option, 23
openstack-baremetal-node-bios-setting-show
    command line option, 24
openstack-baremetal-node-boot-device-show
    command line option, 25
openstack-baremetal-node-console-show
    command line option, 27

```

openstack-baremetal-node-create  
     command line option, 28  
 openstack-baremetal-node-list  
     command line option, 32  
 openstack-baremetal-node-passthru-haste <name>  
     command line option, 35  
 openstack-baremetal-node-show  
     command line option, 42  
 openstack-baremetal-node-trait-list  
     command line option, 43  
 openstack-baremetal-node-validate  
     command line option, 47  
 openstack-baremetal-node-vif-list  
     command line option, 49  
 openstack-baremetal-port-create  
     command line option, 50  
 openstack-baremetal-port-group-create  
     command line option, 51  
 openstack-baremetal-port-group-list  
     command line option, 53  
 openstack-baremetal-port-group-show  
     command line option, 55  
 openstack-baremetal-port-list  
     command line option, 56  
 openstack-baremetal-port-show  
     command line option, 58  
 openstack-baremetal-volume-connector-openstack  
     command line option, 60  
 openstack-baremetal-volume-connector-openstack  
     command line option, 61  
 openstack-baremetal-volume-connector-openstack  
     command line option, 63  
 openstack-baremetal-volume-target-create  
     command line option, 64  
 openstack-baremetal-volume-target-list  
     command line option, 66  
 openstack-baremetal-volume-target-show  
     command line option, 67  
 --mode <MODE>  
 openstack-baremetal-port-group-create  
     command line option, 52  
 openstack-baremetal-port-group-set  
     command line option, 54  
 --name  
 openstack-baremetal-allocation-unset  
     command line option, 12  
 openstack-baremetal-node-unset  
     command line option, 45  
 openstack-baremetal-port-group-unset  
     command line option, 55  
 --name <NAME>  
 openstack-baremetal-allocation-create  
     command line option, 9  
 openstack-baremetal-port-group-create  
     command line option, 52  
 openstack-baremetal-allocation-set  
     command line option, 11  
 openstack-baremetal-node-create  
     command line option, 29  
 openstack-baremetal-node-set  
     command line option, 39  
 openstack-baremetal-port-group-set  
     command line option, 54  
 --network-data  
 openstack-baremetal-node-unset  
     command line option, 45  
 --network-data <network data>  
 openstack-baremetal-node-create  
     command line option, 29  
 openstack-baremetal-node-set  
     command line option, 40  
 --network-interface  
 openstack-baremetal-node-unset  
     command line option, 46  
 --network-interface  
     <network\_interface>  
 openstack-baremetal-node-create  
     command line option, 29  
 openstack-baremetal-node-set  
     command line option, 40  
 --no-maintenance  
 openstack-baremetal-node-list  
     command line option, 33  
 --no-retired  
 openstack-baremetal-node-list  
     command line option, 33  
 --node <NODE>  
 openstack-baremetal-allocation-create  
     command line option, 9  
 --node <node>  
 openstack-baremetal-allocation-list  
     command line option, 10  
 openstack-baremetal-port-group-list  
     command line option, 53  
 openstack-baremetal-port-list  
     command line option, 57  
 openstack-baremetal-volume-connector-list  
     command line option, 62  
 openstack-baremetal-volume-target-list  
     command line option, 66  
 --node <uuid>

openstack-baremetal-port-create  
     command line option, 50

openstack-baremetal-port-group-create  
     command line option, 51

openstack-baremetal-port-group-set  
     command line option, 54

openstack-baremetal-port-set  
     command line option, 57

openstack-baremetal-volume-connector-create  
     command line option, 60

openstack-baremetal-volume-connector-list  
     command line option, 62

openstack-baremetal-volume-target-create  
     command line option, 64

openstack-baremetal-volume-target-set  
     command line option, 67

--noindent

openstack-baremetal-allocation-create  
     command line option, 8

openstack-baremetal-allocation-list  
     command line option, 10

openstack-baremetal-allocation-show  
     command line option, 11

openstack-baremetal-chassis-create  
     command line option, 13

openstack-baremetal-chassis-list  
     command line option, 14

openstack-baremetal-chassis-show  
     command line option, 15

openstack-baremetal-driver-list  
     command line option, 17

openstack-baremetal-driver-passthru-cache  
     command line option, 18

openstack-baremetal-driver-passthru-list  
     command line option, 19

openstack-baremetal-driver-property-list  
     command line option, 20

openstack-baremetal-driver-raid-property-list  
     command line option, 21

openstack-baremetal-driver-show  
     command line option, 22

openstack-baremetal-node-bios-setting-create  
     command line option, 23

openstack-baremetal-node-bios-setting-list  
     command line option, 24

openstack-baremetal-node-boot-device-show  
     command line option, 25

openstack-baremetal-node-console-show  
     command line option, 27

openstack-baremetal-node-create  
     command line option, 28

openstack-baremetal-node-list  
     command line option, 32

openstack-baremetal-node-passthru-list  
     command line option, 35

openstack-baremetal-node-show  
     command line option, 42

openstack-baremetal-node-trait-list  
     command line option, 43

openstack-baremetal-node-validate  
     command line option, 47

openstack-baremetal-node-vif-list  
     command line option, 49

openstack-baremetal-port-create  
     command line option, 50

openstack-baremetal-port-group-create  
     command line option, 51

openstack-baremetal-port-group-list  
     command line option, 53

openstack-baremetal-port-group-show  
     command line option, 55

openstack-baremetal-port-list  
     command line option, 56

openstack-baremetal-port-show  
     command line option, 58

openstack-baremetal-volume-connector-create  
     command line option, 60

openstack-baremetal-volume-connector-list  
     command line option, 61

openstack-baremetal-volume-connector-show  
     command line option, 63

openstack-baremetal-volume-target-create  
     command line option, 64

openstack-baremetal-volume-target-list  
     command line option, 66

openstack-baremetal-volume-target-show  
     command line option, 67

--owner

openstack-baremetal-node-unset  
     command line option, 46

--owner <OWNER>

openstack-baremetal-allocation-create  
     command line option, 9

--owner <owner>

openstack-baremetal-allocation-list  
     command line option, 10

openstack-baremetal-node-create  
     command line option, 30

openstack-baremetal-node-list  
     command line option, 33

openstack-baremetal-node-set  
     command line option, 41

---

```

--persistent
  openstack-baremetal-node-boot-device-set
    command line option, 25
--physical-network
  openstack-baremetal-port-unset
    command line option, 59
--physical-network <physical
  network>
  openstack-baremetal-port-create
    command line option, 50
  openstack-baremetal-port-set
    command line option, 58
--port-group
  openstack-baremetal-port-unset
    command line option, 59
--port-group <port group>
  openstack-baremetal-port-list
    command line option, 57
--port-group <uuid>
  openstack-baremetal-port-create
    command line option, 50
  openstack-baremetal-port-set
    command line option, 57
--port-uuid <port-uuid>
  openstack-baremetal-node-vif-attach
    command line option, 48
--power-interface
  openstack-baremetal-node-unset
    command line option, 46
--power-interface
  <power_interface>
  openstack-baremetal-node-create
    command line option, 29
  openstack-baremetal-node-set
    command line option, 40
--power-timeout <power-timeout>
  openstack-baremetal-node-power-off
    command line option, 36
  openstack-baremetal-node-power-on
    command line option, 36
  openstack-baremetal-node-reboot
    command line option, 37
--prefix <PREFIX>
  openstack-baremetal-allocation-create
    command line option, 8
  openstack-baremetal-allocation-show
    command line option, 11
  openstack-baremetal-chassis-create
    command line option, 13
  openstack-baremetal-chassis-show
    command line option, 15
  openstack-baremetal-driver-passthru-call
    command line option, 18
  openstack-baremetal-driver-show
    command line option, 22
  openstack-baremetal-node-bios-setting-show
    command line option, 24
  openstack-baremetal-node-boot-device-show
    command line option, 25
  openstack-baremetal-node-console-show
    command line option, 27
  openstack-baremetal-node-create
    command line option, 28
  openstack-baremetal-node-show
    command line option, 42
  openstack-baremetal-port-create
    command line option, 50
  openstack-baremetal-port-group-create
    command line option, 51
  openstack-baremetal-port-group-show
    command line option, 55
  openstack-baremetal-port-show
    command line option, 58
  openstack-baremetal-volume-connector-create
    command line option, 60
  openstack-baremetal-volume-connector-show
    command line option, 63
  openstack-baremetal-volume-target-create
    command line option, 64
  openstack-baremetal-volume-target-show
    command line option, 67
--print-empty
  openstack-baremetal-allocation-create
    command line option, 8
  openstack-baremetal-allocation-list
    command line option, 10
  openstack-baremetal-allocation-show
    command line option, 12
  openstack-baremetal-chassis-create
    command line option, 13
  openstack-baremetal-chassis-list
    command line option, 14
  openstack-baremetal-chassis-show
    command line option, 16
  openstack-baremetal-driver-list
    command line option, 17
  openstack-baremetal-driver-passthru-call
    command line option, 18
  openstack-baremetal-driver-passthru-list
    command line option, 19
  openstack-baremetal-driver-property-list
    command line option, 20

```

```

openstack-baremetal-driver-raid-properties openstack-baremetal-node-create
  command line option, 21 command line option, 29
openstack-baremetal-driver-show openstack-baremetal-node-set
  command line option, 22 command line option, 41
openstack-baremetal-node-bios-settings openstack-baremetal-port-group-create
  command line option, 24 command line option, 52
openstack-baremetal-node-bios-settings openstack-baremetal-port-group-set
  command line option, 24 command line option, 54
openstack-baremetal-node-boot-device-show openstack-baremetal-volume-target-create
  command line option, 26 command line option, 64
openstack-baremetal-node-console-show openstack-baremetal-volume-target-set
  command line option, 27 command line option, 67
openstack-baremetal-node-create --property <key>
  command line option, 29 openstack-baremetal-node-unset
openstack-baremetal-node-list command line option, 45
  command line option, 32 openstack-baremetal-port-group-unset
openstack-baremetal-node-passthru-list command line option, 56
  command line option, 36 openstack-baremetal-volume-target-unset
openstack-baremetal-node-show command line option, 68
  command line option, 42 --protected
openstack-baremetal-node-trait-list openstack-baremetal-node-set
  command line option, 43 command line option, 41
openstack-baremetal-node-validate openstack-baremetal-node-unset
  command line option, 47 command line option, 46
openstack-baremetal-node-vif-list-protected-reason
  command line option, 49 openstack-baremetal-node-unset
openstack-baremetal-port-create command line option, 46
  command line option, 50 --protected-reason
openstack-baremetal-port-group-create <protected_reason>
  command line option, 51 openstack-baremetal-node-set
openstack-baremetal-port-group-list command line option, 41
  command line option, 53 --provision-state <provision
openstack-baremetal-port-group-show state>
  command line option, 55 openstack-baremetal-node-list
openstack-baremetal-port-list command line option, 33
  command line option, 56 --pxe-disabled
openstack-baremetal-port-show openstack-baremetal-port-set
  command line option, 59 command line option, 58
openstack-baremetal-volume-connector-pxe-enabled
  command line option, 60 openstack-baremetal-port-set
openstack-baremetal-volume-connector-list command line option, 58
  command line option, 61 --pxe-enabled <boolean>
openstack-baremetal-volume-connector-show openstack-baremetal-port-create
  command line option, 63 command line option, 50
openstack-baremetal-volume-target-create openstack-baremetal-allocation-list
  command line option, 64 command line option, 10
openstack-baremetal-volume-target-list command line option, 66
  command line option, 66 openstack-baremetal-chassis-list
openstack-baremetal-volume-target-show command line option, 14
  command line option, 68 openstack-baremetal-driver-list
--property <key=value> command line option, 17

```

openstack-baremetal-driver-passthrough-list  
     command line option, 19

openstack-baremetal-driver-property-reset  
     command line option, 20

openstack-baremetal-driver-raid-property-list  
     command line option, 21

openstack-baremetal-node-bios-settings  
     command line option, 23

openstack-baremetal-node-list  
     command line option, 32

openstack-baremetal-node-passthrough-list  
     command line option, 35

openstack-baremetal-node-trait-list  
     command line option, 43

openstack-baremetal-node-validate  
     command line option, 47

openstack-baremetal-node-vif-list  
     command line option, 49

openstack-baremetal-port-group-list  
     command line option, 53

openstack-baremetal-port-list  
     command line option, 56

openstack-baremetal-volume-connector-list  
     command line option, 61

openstack-baremetal-volume-target-list  
     command line option, 65

--raid-interface  
     openstack-baremetal-node-unset  
         command line option, 46

--raid-interface <raid\_interface>  
     openstack-baremetal-node-create  
         command line option, 30

--reason <reason>  
     openstack-baremetal-node-maintenance-set  
         command line option, 34

--rescue-interface  
     openstack-baremetal-node-unset  
         command line option, 46

--rescue-interface <rescue\_interface>  
     openstack-baremetal-node-create  
         command line option, 30

--rescue-password  
     <rescue-password>  
     openstack-baremetal-node-rescue  
         command line option, 38

--reset-bios-interface  
     openstack-baremetal-node-set  
         command line option, 39

    openstack-baremetal-node-set  
         openstack-baremetal-node-set  
         command line option, 40

    --reset-console-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-deploy-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-inspect-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-interfaces  
     openstack-baremetal-node-set  
         command line option, 41

    --reset-management-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-network-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-power-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-raid-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-rescue-interface  
     openstack-baremetal-node-set  
         command line option, 40

    --reset-storage-interface  
     openstack-baremetal-node-set  
         command line option, 41

    --reset-vendor-interface  
     openstack-baremetal-node-set  
         command line option, 41

    --resource-class  
     openstack-baremetal-node-unset  
         command line option, 45

    --resource-class <RESOURCE\_CLASS>  
     openstack-baremetal-allocation-create  
         command line option, 8

    --resource-class <resource class>  
     openstack-baremetal-node-list  
         command line option, 33

    --resource-class <resource\_class>  
     openstack-baremetal-allocation-list  
         command line option, 10

    openstack-baremetal-node-create



```

        command line option, 30
openstack-baremetal-node-set
    command line option, 41
--retired
    openstack-baremetal-node-list
        command line option, 33
    openstack-baremetal-node-set
        command line option, 41
    openstack-baremetal-node-unset
        command line option, 46
--retired-reason
    openstack-baremetal-node-unset
        command line option, 46
--retired-reason <retired_reason>
    openstack-baremetal-node-set
        command line option, 41
--soft
    openstack-baremetal-node-power-off
        command line option, 36
    openstack-baremetal-node-reboot
        command line option, 37
--sort <key>[:<direction>]
    openstack-baremetal-allocation-list
        command line option, 10
    openstack-baremetal-chassis-list
        command line option, 15
    openstack-baremetal-node-list
        command line option, 33
    openstack-baremetal-port-group-list
        command line option, 53
    openstack-baremetal-port-list
        command line option, 57
    openstack-baremetal-volume-connector-list
        command line option, 62
    openstack-baremetal-volume-target-list
        command line option, 66
--sort-column SORT_COLUMN
    openstack-baremetal-allocation-list
        command line option, 10
    openstack-baremetal-chassis-list
        command line option, 14
    openstack-baremetal-driver-list
        command line option, 17
    openstack-baremetal-driver-passthru-list
        command line option, 19
    openstack-baremetal-driver-property-list
        command line option, 20
    openstack-baremetal-driver-raid-property-list
        command line option, 21
    openstack-baremetal-node-bios-setting-list
        command line option, 24
    openstack-baremetal-node-list
        command line option, 32
    openstack-baremetal-node-passthru-list
        command line option, 36
    openstack-baremetal-node-trait-list
        command line option, 43
    openstack-baremetal-node-validate
        command line option, 47
    openstack-baremetal-node-vif-list
        command line option, 49
    openstack-baremetal-port-group-list
        command line option, 53
    openstack-baremetal-port-list
        command line option, 56
    openstack-baremetal-volume-connector-list
        command line option, 61
    openstack-baremetal-volume-target-list
        command line option, 66
--state <state>
    openstack-baremetal-allocation-list
        command line option, 10
--storage-interface
    openstack-baremetal-node-unset
        command line option, 46
    openstack-baremetal-node-create
        command line option, 30
    openstack-baremetal-node-set
        command line option, 40
--support-standalone-ports
    openstack-baremetal-port-group-create
        command line option, 52
    openstack-baremetal-port-group-set
        command line option, 54
--supported
    openstack-baremetal-node-boot-device-show
        command line option, 26
--target-raid-config
    openstack-baremetal-node-unset
        command line option, 45
    openstack-baremetal-node-set
        command line option, 41
--traits <TRAITS>
    openstack-baremetal-allocation-create
        command line option, 9
--type <type>
    openstack-baremetal-driver-list
        command line option, 17

```



openstack-baremetal-volume-connector create  
     command line option, 60  
 openstack-baremetal-volume-connector delete  
     command line option, 62  
 --type <volume type>  
     openstack-baremetal-volume-target-create  
         command line option, 64  
     openstack-baremetal-volume-target-set  
         command line option, 67  
 --unassociated  
     openstack-baremetal-node-list  
         command line option, 33  
 --unsupport-standalone-ports  
     openstack-baremetal-port-group-create  
         command line option, 52  
     openstack-baremetal-port-group-set  
         command line option, 54  
 --uuid <UUID>  
     openstack-baremetal-allocation-create  
         command line option, 9  
     openstack-baremetal-port-group-create  
         command line option, 52  
 --uuid <uuid>  
     openstack-baremetal-chassis-create  
         command line option, 13  
     openstack-baremetal-node-create  
         command line option, 29  
     openstack-baremetal-port-create  
         command line option, 50  
     openstack-baremetal-volume-connector create  
         command line option, 60  
     openstack-baremetal-volume-target-create  
         command line option, 65  
 --vendor-interface  
     openstack-baremetal-node-unset  
         command line option, 46  
 --vendor-interface <vendor\_interface>  
     openstack-baremetal-node-create  
         command line option, 30  
     openstack-baremetal-node-set  
         command line option, 41  
 --vif-info <key=value>  
     openstack-baremetal-node-vif-attach  
         command line option, 48  
 --volume-id <volume id>  
     openstack-baremetal-volume-target-create  
         command line option, 65  
     openstack-baremetal-volume-target-set  
         command line option, 67  
 --wait <time-out>

openstack-baremetal-node-boot-device-show  
     command line option, 25

openstack-baremetal-node-console-show  
     command line option, 27

openstack-baremetal-node-create  
     command line option, 28

openstack-baremetal-node-list  
     command line option, 32

openstack-baremetal-node-passthru-list  
     command line option, 35

openstack-baremetal-node-show  
     command line option, 42

openstack-baremetal-node-trait-list  
     command line option, 43

openstack-baremetal-node-validate  
     command line option, 47

openstack-baremetal-node-vif-list  
     command line option, 49

openstack-baremetal-port-create  
     command line option, 50

openstack-baremetal-port-group-create  
     command line option, 51

openstack-baremetal-port-group-list  
     command line option, 53

openstack-baremetal-port-group-show  
     command line option, 55

openstack-baremetal-port-list  
     command line option, 56

openstack-baremetal-port-show  
     command line option, 58

openstack-baremetal-volume-connector-create  
     command line option, 60

openstack-baremetal-volume-connector-list  
     command line option, 61

openstack-baremetal-volume-connector-show  
     command line option, 63

openstack-baremetal-volume-target-create  
     command line option, 64

openstack-baremetal-volume-target-list  
     command line option, 65

openstack-baremetal-volume-target-show  
     command line option, 67

-f <FORMATTER>

openstack-baremetal-allocation-create  
     command line option, 8

openstack-baremetal-allocation-list  
     command line option, 10

openstack-baremetal-allocation-show  
     command line option, 11

openstack-baremetal-chassis-create  
     command line option, 13

openstack-baremetal-chassis-list  
     command line option, 14

openstack-baremetal-chassis-show  
     command line option, 15

openstack-baremetal-driver-list  
     command line option, 17

openstack-baremetal-driver-passthru-call  
     command line option, 18

openstack-baremetal-driver-passthru-list  
     command line option, 19

openstack-baremetal-driver-property-list  
     command line option, 20

openstack-baremetal-driver-raid-property-list  
     command line option, 21

openstack-baremetal-driver-show  
     command line option, 22

openstack-baremetal-node-bios-setting-list  
     command line option, 23

openstack-baremetal-node-bios-setting-show  
     command line option, 24

openstack-baremetal-node-boot-device-show  
     command line option, 25

openstack-baremetal-node-console-show  
     command line option, 27

openstack-baremetal-node-create  
     command line option, 28

openstack-baremetal-node-list  
     command line option, 32

openstack-baremetal-node-passthru-list  
     command line option, 35

openstack-baremetal-node-show  
     command line option, 42

openstack-baremetal-node-trait-list  
     command line option, 43

openstack-baremetal-node-validate  
     command line option, 47

openstack-baremetal-node-vif-list  
     command line option, 48

openstack-baremetal-port-create  
     command line option, 49

openstack-baremetal-port-group-create  
     command line option, 51

openstack-baremetal-port-group-list  
     command line option, 53

openstack-baremetal-port-group-show  
     command line option, 55

openstack-baremetal-port-list  
     command line option, 56

openstack-baremetal-port-show  
     command line option, 58

openstack-baremetal-volume-connector-create  
     command line option, 60

openstack-baremetal-volume-connector-list  
     command line option, 61

openstack-baremetal-volume-connector-show  
     command line option, 63

openstack-baremetal-volume-target-create  
     command line option, 64

openstack-baremetal-volume-target-list  
     command line option, 65

openstack-baremetal-volume-target-show  
     command line option, 67

- command line option, 60
  - openstack-baremetal-volume-connector-attach
    - command line option, 61
  - openstack-baremetal-volume-connector-show
    - command line option, 63
  - openstack-baremetal-volume-target-create
    - command line option, 64
  - openstack-baremetal-volume-target-erase
    - command line option, 65
  - openstack-baremetal-volume-target-show
    - command line option, 67
  - l <key=value>
    - openstack-baremetal-port-create
      - command line option, 50
- A**
- AbortBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 100
  - add\_hook() (*ironic-client.common.apiclient.base.HookableMixin* class method), 75
  - add\_trait() (*ironic-client.v1.node.NodeManager* method), 128
  - address
    - openstack-baremetal-port-create
      - command line option, 50
  - AddTraitBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 100
  - AdoptBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 101
  - allocation
    - openstack-baremetal-allocation-delete
      - command line option, 9
    - openstack-baremetal-allocation-set
      - command line option, 11
    - openstack-baremetal-allocation-unset
      - command line option, 12
  - Allocation (class in *ironic-client.v1.allocation*), 117
  - AllocationManager (class in *ironic-client.v1.allocation*), 117
  - AmbiguousAuthSystem (in module *ironic-client.exc*), 145
  - AmbiguousAuthSystem, 145
  - AmbiguousEndpoints, 76
  - App (class in *ironicclient.shell*), 146
  - args\_array\_to\_dict() (in module *ironic-client.common.utils*), 90
  - args\_array\_to\_patch() (in module *ironic-client.common.utils*), 90
  - AuthorizationFailure, 76
  - AuthPluginOptionsMissing, 76
  - AuthSystemNotFound, 76
- B**
- BadGateway, 76
  - BadRequest, 77
  - baremetal() (*ironicclient.shell.ClientManager* property), 146
  - baremetal\_introspection() (*ironic-client.shell.ClientManager* property), 146
  - BaseManager (class in *ironic-client.common.apiclient.base*), 74
  - BIOSSettingShowBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 101
  - bool\_argument\_value() (in module *ironic-client.common.utils*), 90
  - BootdeviceSetBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 101
  - BootdeviceShowBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 101
  - build\_option\_parser() (in module *ironic-client.osc.plugin*), 117
  - build\_option\_parser() (*ironic-client.shell.App* method), 146
  - build\_url() (*ironic-client.common.apiclient.base.CrudManager* method), 74
- C**
- chassis
    - openstack-baremetal-chassis-delete
      - command line option, 13
    - openstack-baremetal-chassis-set
      - command line option, 15
    - openstack-baremetal-chassis-show
      - command line option, 16
    - openstack-baremetal-chassis-unset
      - command line option, 16
  - Chassis (class in *ironicclient.v1.chassis*), 120
  - ChassisManager (class in *ironic-client.v1.chassis*), 120
  - check\_empty\_arg() (in module *ironic-client.common.utils*), 91
  - check\_for\_invalid\_fields() (in module *ironicclient.common.utils*), 91
  - CleanBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 101
  - Client (class in *ironicclient.v1.client*), 122
  - Client() (in module *ironicclient.client*), 144

ClientException, 77

ClientManager (class in *ironicclient.shell*), 146

collection\_key (in module *ironicclient.common.apiclient.base.CrudManager* attribute), 74

CommandError, 77

CommandManager (class in *ironicclient.shell*), 146

common\_filters() (in module *ironicclient.common.utils*), 91

common\_params\_for\_list() (in module *ironicclient.common.utils*), 91

Conductor (class in *ironicclient.v1.conductor*), 122

ConductorManager (class in *ironicclient.v1.conductor*), 122

Conflict, 77

ConnectionError, 78

ConnectionRefused, 78

ConsoleDisableBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 102

ConsoleEnableBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 102

ConsoleShowBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 102

convert\_list\_props\_to\_comma\_separated() (in module *ironicclient.common.utils*), 91

create() (in module *ironicclient.common.apiclient.base.CrudManager* method), 74

create() (in module *ironicclient.common.base.CreateManager* method), 88

create\_chassis() (in module *ironicclient.v1.create\_resources*), 123

create\_nodes() (in module *ironicclient.v1.create\_resources*), 123

create\_portgroups() (in module *ironicclient.v1.create\_resources*), 123

create\_ports() (in module *ironicclient.v1.create\_resources*), 124

create\_resources() (in module *ironicclient.v1.create\_resources*), 124

create\_single\_chassis() (in module *ironicclient.v1.create\_resources*), 124

create\_single\_handler() (in module *ironicclient.v1.create\_resources*), 124

create\_single\_node() (in module *ironicclient.v1.create\_resources*), 125

create\_single\_port() (in module *ironicclient.v1.create\_resources*), 125

create\_single\_portgroup() (in module *ironicclient.v1.create\_resources*), 125

CreateBaremetal (class in *ironicclient.osc.v1.baremetal\_create*), 97

CreateBaremetalAllocation (class in *ironicclient.osc.v1.baremetal\_allocation*), 93

CreateBaremetalChassis (class in *ironicclient.osc.v1.baremetal\_chassis*), 95

CreateBaremetalDeployTemplate (class in *ironicclient.osc.v1.baremetal\_deploy\_template*), 97

CreateBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 102

CreateBaremetalPort (class in *ironicclient.osc.v1.baremetal\_port*), 110

CreateBaremetalPortGroup (class in *ironicclient.osc.v1.baremetal\_portgroup*), 112

CreateBaremetalVolumeConnector (class in *ironicclient.osc.v1.baremetal\_volume\_connector*), 114

CreateBaremetalVolumeTarget (class in *ironicclient.osc.v1.baremetal\_volume\_target*), 115

CreateManager (class in *ironicclient.common.base*), 88

CrudManager (class in *ironicclient.common.apiclient.base*), 74

current\_api\_version() (in module *ironicclient.v1.client* Client property), 122

## D

define\_command() (in module *ironicclient.common.utils*), 91

define\_commands\_from\_module() (in module *ironicclient.common.utils*), 92

delete() (in module *ironicclient.common.apiclient.base.CrudManager* method), 74

delete() (in module *ironicclient.v1.allocation.AllocationManager* method), 118

delete() (in module *ironicclient.v1.chassis.ChassisManager* method), 120

delete() (*ironic-client.v1.deploy\_template.DeployTemplateManager* method), 126

delete() (*ironic-client.v1.driver.DriverManager* method), 127

delete() (*ironicclient.v1.node.NodeManager* method), 129

delete() (*ironicclient.v1.port.PortManager* method), 138

delete() (*ironic-client.v1.portgroup.PortgroupManager* method), 139

delete() (*ironic-client.v1.volume\_connector.VolumeConnectorManager* method), 142

delete() (*ironic-client.v1.volume\_target.VolumeTargetManager* method), 143

DeleteBaremetalAllocation (class in *ironic-client.osc.v1.baremetal\_allocation*), 93

DeleteBaremetalChassis (class in *ironic-client.osc.v1.baremetal\_chassis*), 95

DeleteBaremetalDeployTemplate (class in *ironic-client.osc.v1.baremetal\_deploy\_template*), 97

DeleteBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 103

DeleteBaremetalPort (class in *ironic-client.osc.v1.baremetal\_port*), 110

DeleteBaremetalPortGroup (class in *ironicclient.osc.v1.baremetal\_portgroup*), 112

DeleteBaremetalVolumeConnector (class in *ironic-client.osc.v1.baremetal\_volume\_connector*), 114

DeleteBaremetalVolumeTarget (class in *ironic-client.osc.v1.baremetal\_volume\_target*), 115

DeployBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 103

DeployTemplate (class in *ironic-client.v1.deploy\_template*), 126

DeployTemplateManager (class in *ironic-client.v1.deploy\_template*), 126

device

openstack-baremetal-node-boot-device-set command line option, 25

driver

openstack-baremetal-driver-passthru-call command line option, 18

openstack-baremetal-driver-passthru-list command line option, 19

openstack-baremetal-driver-property-list command line option, 20

openstack-baremetal-driver-raid-property-list command line option, 21

openstack-baremetal-driver-show command line option, 22

Driver (class in *ironicclient.v1.driver*), 127

DriverManager (class in *ironic-client.v1.driver*), 127

**E**

EndpointException, 78

EndpointNotFound, 78

Event (class in *ironicclient.v1.events*), 128

EventManager (class in *ironicclient.v1.events*), 128

ExpectationFailed, 78

Extension (class in *ironic-client.common.apiclient.base*), 75

**F**

FIELDS (*ironicclient.v1.resource\_fields.Resource* attribute), 142

fields() (*ironic-client.v1.resource\_fields.Resource* property), 142

file

openstack-baremetal-create command line option, 16

find() (*ironic-client.common.apiclient.base.CrudManager* method), 74

find() (*ironic-client.common.apiclient.base.ManagerWithFind* method), 75

findall() (*ironic-client.common.apiclient.base.ManagerWithFind* method), 75

Forbidden, 78

from\_response() (in module *ironic-client.common.apiclient.exceptions*), 88

from\_response() (in module *ironic-client.exc*), 146



G

GatewayTimeout, 79		get_parser()	(ironic-client.osc.v1.baremetal_allocation.DeleteBaremetalAllocation method), 93
get () (ironicclient.common.apiclient.base.CrudManager method), 74		get_parser()	(ironic-client.osc.v1.baremetal_allocation.ListBaremetalAllocation method), 94
get () (ironicclient.common.apiclient.base.ResourceManager method), 76		get_parser()	(ironic-client.osc.v1.baremetal_allocation.SetBaremetalAllocation method), 94
get () (ironicclient.v1.allocation.AllocationManager method), 118		get_parser()	(ironic-client.osc.v1.baremetal_allocation.ShowBaremetalAllocation method), 94
get () (ironicclient.v1.chassis.ChassisManager method), 120		get_parser()	(ironic-client.osc.v1.baremetal_allocation.UnsetBaremetalAllocation method), 94
get () (ironicclient.v1.conductor.ConductorManager method), 122		get_parser()	(ironic-client.osc.v1.baremetal_chassis.CreateBaremetalChassis method), 95
get () (ironicclient.v1.deploy_template.DeployTemplateManager method), 126		get_parser()	(ironic-client.osc.v1.baremetal_chassis.DeleteBaremetalChassis method), 95
get () (ironicclient.v1.driver.DriverManager method), 127		get_parser()	(ironic-client.osc.v1.baremetal_chassis.ListBaremetalChassis method), 95
get () (ironicclient.v1.node.NodeManager method), 129		get_parser()	(ironic-client.osc.v1.baremetal_chassis.SetBaremetalChassis method), 95
get () (ironicclient.v1.port.PortManager method), 138		get_parser()	(ironic-client.osc.v1.baremetal_chassis.ShowBaremetalChassis method), 96
get () (ironicclient.v1.portgroup.PortgroupManager method), 139		get_parser()	(ironic-client.osc.v1.baremetal_chassis.UnsetBaremetalChassis method), 96
get () (ironicclient.v1.volume_connector.VolumeConnectorManager method), 142		get_parser()	(ironic-client.osc.v1.baremetal_conductor.ListBaremetalConductors method), 96
get () (ironicclient.v1.volume_target.VolumeTargetManager method), 143		get_parser()	(ironic-client.osc.v1.baremetal_create.CreateBaremetal method), 97
get_bios_setting () (ironic-client.v1.node.NodeManager method), 129		get_parser()	(ironic-client.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplate method), 97
get_boot_device () (ironic-client.v1.node.NodeManager method), 129		get_parser()	(ironic-client.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate method), 97
get_by_address () (ironic-client.v1.port.PortManager method), 138		get_parser()	(ironic-client.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplates method), 98
get_by_address () (ironic-client.v1.portgroup.PortgroupManager method), 139		get_parser()	
get_by_instance_uuid () (ironic-client.v1.node.NodeManager method), 129		get_parser()	
get_client () (in module ironicclient.client), 144		get_parser()	
get_console () (ironic-client.v1.node.NodeManager method), 129		get_parser()	
get_from_stdin () (in module ironicclient.common.utils), 92		get_parser()	
get_parser () (ironic-client.osc.v1.baremetal_allocation.CreateBaremetalAllocation method), 93		get_parser()	

get\_parser() (ironic- client.osc.v1.baremetal\_deploy\_template.SetBaremetalDeployTemplate method), 98 (ironic- client.osc.v1.baremetal\_node.CreateBaremetalNode method), 103

get\_parser() (ironic- client.osc.v1.baremetal\_deploy\_template.ShowBaremetalDeployTemplate method), 98 (ironic- client.osc.v1.baremetal\_node.DeleteBaremetalNode method), 103

get\_parser() (ironic- client.osc.v1.baremetal\_deploy\_template.UnsetBaremetalDeployTemplate method), 98 (ironic- client.osc.v1.baremetal\_node.DeployBaremetalNode method), 103

get\_parser() (ironic- client.osc.v1.baremetal\_driver.ListBaremetalDriver method), 99 (ironic- client.osc.v1.baremetal\_node.InjectNmiBaremetalNode method), 103

get\_parser() (ironic- client.osc.v1.baremetal\_driver.ListBaremetalDriverPart method), 99 (ironic- client.osc.v1.baremetal\_node.ListBaremetalNode method), 104

get\_parser() (ironic- client.osc.v1.baremetal\_driver.ListBaremetalDriverProperties method), 99 (ironic- client.osc.v1.baremetal\_node.ListBIOSSettingBaremetalNode method), 104

get\_parser() (ironic- client.osc.v1.baremetal\_driver.PassthruCallBaremetalDriver method), 99 (ironic- client.osc.v1.baremetal\_node.ListTraitsBaremetalNode method), 104

get\_parser() (ironic- client.osc.v1.baremetal\_driver.PassthruListBaremetalDriver method), 100 (ironic- client.osc.v1.baremetal\_node.MaintenanceSetBaremetalNode method), 105

get\_parser() (ironic- client.osc.v1.baremetal\_driver.ShowBaremetalDriver method), 100 (ironic- client.osc.v1.baremetal\_node.MaintenanceUnsetBaremetalNode method), 105

get\_parser() (ironic- client.osc.v1.baremetal\_node.AddTraitBaremetalNode method), 100 (ironic- client.osc.v1.baremetal\_node.PassthruCallBaremetalNode method), 105

get\_parser() (ironic- client.osc.v1.baremetal\_node.BIOSSettingShowBaremetalNode method), 101 (ironic- client.osc.v1.baremetal\_node.PassthruListBaremetalNode method), 105

get\_parser() (ironic- client.osc.v1.baremetal\_node.BootdeviceSetBaremetalNode method), 101 (ironic- client.osc.v1.baremetal\_node.PowerBaremetalNode method), 106

get\_parser() (ironic- client.osc.v1.baremetal\_node.BootdeviceShowBaremetalNode method), 101 (ironic- client.osc.v1.baremetal\_node.PowerOffBaremetalNode method), 106

get\_parser() (ironic- client.osc.v1.baremetal\_node.CleanBaremetalNode method), 102 (ironic- client.osc.v1.baremetal\_node.ProvisionStateBaremetalNode method), 107

get\_parser() (ironic- client.osc.v1.baremetal\_node.ConsoleDisableBaremetalNode method), 102 (ironic- client.osc.v1.baremetal\_node.ProvisionStateWithWait method), 107

get\_parser() (ironic- client.osc.v1.baremetal\_node.ConsoleEnableBaremetalNode method), 102 (ironic- client.osc.v1.baremetal\_node.RebootBaremetalNode method), 107

get\_parser() (ironic- client.osc.v1.baremetal\_node.ConsoleShowBaremetalNode method), 102 (ironic- client.osc.v1.baremetal\_node.RebuildBaremetalNode method), 107

get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</i>	<i>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</i>	<i>client.osc.v1.baremetal_portgroup.ListBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.ListBaremetalPortGroup</i>
method), 108	method), 108	method), 112	method), 112
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.RescueBaremetalNode</i>	<i>client.osc.v1.baremetal_node.RescueBaremetalNode</i>	<i>client.osc.v1.baremetal_portgroup.SetBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.SetBaremetalPortGroup</i>
method), 108	method), 108	method), 113	method), 113
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.SetBaremetalNode</i>	<i>client.osc.v1.baremetal_node.SetBaremetalNode</i>	<i>client.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup</i>
method), 108	method), 108	method), 113	method), 113
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.ShowBaremetalNode</i>	<i>client.osc.v1.baremetal_node.ShowBaremetalNode</i>	<i>client.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup</i>
method), 108	method), 108	method), 113	method), 113
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.UnsetBaremetalNode</i>	<i>client.osc.v1.baremetal_node.UnsetBaremetalNode</i>	<i>client.osc.v1.baremetal_volume_connector.CreateBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_connector.CreateBaremetalVolume</i>
method), 109	method), 109	method), 114	method), 114
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.ValidateBaremetalNode</i>	<i>client.osc.v1.baremetal_node.ValidateBaremetalNode</i>	<i>client.osc.v1.baremetal_volume_connector.DeleteBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_connector.DeleteBaremetalVolume</i>
method), 109	method), 109	method), 114	method), 114
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.VifAttachBaremetalNode</i>	<i>client.osc.v1.baremetal_node.VifAttachBaremetalNode</i>	<i>client.osc.v1.baremetal_volume_connector.ListBaremetalVolumes</i>	<i>client.osc.v1.baremetal_volume_connector.ListBaremetalVolumes</i>
method), 109	method), 109	method), 114	method), 114
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.VifDetachBaremetalNode</i>	<i>client.osc.v1.baremetal_node.VifDetachBaremetalNode</i>	<i>client.osc.v1.baremetal_volume_connector.SetBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_connector.SetBaremetalVolume</i>
method), 110	method), 110	method), 114	method), 114
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_node.VifListBaremetalNode</i>	<i>client.osc.v1.baremetal_node.VifListBaremetalNode</i>	<i>client.osc.v1.baremetal_volume_connector.ShowBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_connector.ShowBaremetalVolume</i>
method), 110	method), 110	method), 115	method), 115
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.CreateBaremetalPort</i>	<i>client.osc.v1.baremetal_port.CreateBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_connector.UnsetBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_connector.UnsetBaremetalVolume</i>
method), 110	method), 110	method), 115	method), 115
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.DeleteBaremetalPort</i>	<i>client.osc.v1.baremetal_port.DeleteBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_target.CreateBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_target.CreateBaremetalVolume</i>
method), 111	method), 111	method), 115	method), 115
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.ListBaremetalPort</i>	<i>client.osc.v1.baremetal_port.ListBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_target.DeleteBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_target.DeleteBaremetalVolume</i>
method), 111	method), 111	method), 115	method), 115
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.SetBaremetalPort</i>	<i>client.osc.v1.baremetal_port.SetBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_target.ListBaremetalVolumes</i>	<i>client.osc.v1.baremetal_volume_target.ListBaremetalVolumes</i>
method), 111	method), 111	method), 116	method), 116
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.ShowBaremetalPort</i>	<i>client.osc.v1.baremetal_port.ShowBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_target.SetBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_target.SetBaremetalVolume</i>
method), 111	method), 111	method), 116	method), 116
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_port.UnsetBaremetalPort</i>	<i>client.osc.v1.baremetal_port.UnsetBaremetalPort</i>	<i>client.osc.v1.baremetal_volume_target.ShowBaremetalVolumes</i>	<i>client.osc.v1.baremetal_volume_target.ShowBaremetalVolumes</i>
method), 112	method), 112	method), 116	method), 116
get_parser()	(ironic-	get_parser()	(ironic-
<i>client.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_volume_target.UnsetBaremetalVolume</i>	<i>client.osc.v1.baremetal_volume_target.UnsetBaremetalVolume</i>
method), 112	method), 112	method), 116	method), 116
get_parser()	(ironic-	get_server()	(in module ironic-
<i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup</i>	<i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup</i>
method), 112	method), 112	method), 112	method), 90
	get_supported_boot_devices()		(ironic-



*icclient.v1.node.NodeManager* method), 129

`get_traits()` (*ironic-client.v1.node.NodeManager* method), 129

`get_vendor_passthru_methods()` (*ironicclient.v1.driver.DriverManager* method), 127

`get_vendor_passthru_methods()` (*ironicclient.v1.node.NodeManager* method), 129

`getid()` (in module *ironic-client.common.apiclient.base*), 76

`getid()` (in module *ironicclient.common.base*), 89

Gone, 79

## H

`handle_json_arg()` (in module *ironic-client.common.utils*), 92

`handle_json_or_file_arg()` (in module *ironicclient.common.utils*), 92

`head()` (*ironic-client.common.apiclient.base.CrudManager* method), 74

`HelpFormatter` (class in *ironic-client.common.utils*), 90

`HookableMixin` (class in *ironic-client.common.apiclient.base*), 75

`http_status` (*ironic-client.common.apiclient.exceptions.BadGateway* attribute), 77

`http_status` (*ironic-client.common.apiclient.exceptions.BadRequest* attribute), 77

`http_status` (*ironic-client.common.apiclient.exceptions.Conflict* attribute), 78

`http_status` (*ironic-client.common.apiclient.exceptions.ExpectationFailed* attribute), 78

`http_status` (*ironic-client.common.apiclient.exceptions.Forbidden* attribute), 79

`http_status` (*ironic-client.common.apiclient.exceptions.GatewayTimeout* attribute), 79

`http_status` (*ironic-client.common.apiclient.exceptions.Gone* attribute), 79

`http_status` (*ironic-client.common.apiclient.exceptions.HttpError* attribute), 80

`http_status` (*ironic-client.common.apiclient.exceptions.HttpNotImplemented* attribute), 81

`http_status` (*ironic-client.common.apiclient.exceptions.HttpVersionNotSupported* attribute), 81

`http_status` (*ironic-client.common.apiclient.exceptions.InternalServerError* attribute), 82

`http_status` (*ironic-client.common.apiclient.exceptions.LengthRequired* attribute), 82

`http_status` (*ironic-client.common.apiclient.exceptions.MethodNotAllowed* attribute), 82

`http_status` (*ironic-client.common.apiclient.exceptions.MultipleChoices* attribute), 83

`http_status` (*ironic-client.common.apiclient.exceptions.NotAcceptable* attribute), 83

`http_status` (*ironic-client.common.apiclient.exceptions.NotFound* attribute), 84

`http_status` (*ironic-client.common.apiclient.exceptions.PaymentRequired* attribute), 84

`http_status` (*ironic-client.common.apiclient.exceptions.PreconditionFailed* attribute), 84

`http_status` (*ironic-client.common.apiclient.exceptions.ProxyAuthenticationRequired* attribute), 85

`http_status` (*ironic-client.common.apiclient.exceptions.RequestedRangeNotSatisfiable* attribute), 86

`http_status` (*ironic-client.common.apiclient.exceptions.RequestEntityTooLarge* attribute), 85

`http_status` (*ironic-client.common.apiclient.exceptions.RequestTimeout* attribute), 85

`http_status` (*ironic-client.common.apiclient.exceptions.RequestUriTooLong* attribute), 86

`http_status` (*ironic-client.common.apiclient.exceptions.ServiceUnavailable* attribute), 86

`http_status` (*ironic-client.common.apiclient.exceptions.UnprocessableEntity* attribute), 86

*client.common.apiclient.exceptions.Unauthorized* module, 88

*attribute*), 87

*http\_status* (*ironic-client.common.apiclient.exceptions.UnprocessableEntity* module, 74

*attribute*), 87

*http\_status* (*ironic-client.common.apiclient.exceptions.UnprocessableEntity* module, 76

*attribute*), 87

HTTPClientError, 80

HttpError, 80

HttpNotImplemented, 80

HTTPRedirection, 80

HttpServerError, 81

HttpVersionNotSupported, 81

HUMAN\_ID (*ironic-client.common.apiclient.base.Resource* module, 88

*attribute*), 76

human\_id() (*ironic-client.common.apiclient.base.Resource* module, 89

*property*), 76

**I**

id (*ironic-client.common.apiclient.base.Resource* module, 89

*property*), 76

openstack-baremetal-allocation-show (*ironic-client.common.apiclient.base.Resource* module, 89

command line option, 12

openstack-baremetal-port-group-show (*ironic-client.common.apiclient.base.Resource* module, 89

command line option, 55

openstack-baremetal-port-show (*ironic-client.common.apiclient.base.Resource* module, 89

command line option, 59

openstack-baremetal-volume-connection-show (*ironic-client.common.apiclient.base.Resource* module, 89

command line option, 63

openstack-baremetal-volume-target-connection-show (*ironic-client.common.apiclient.base.Resource* module, 89

command line option, 68

initialize\_app() (*ironicclient.shell.App* module, 89

*method*), 146

inject\_nmi() (*ironic-client.v1.node.NodeManager* module, 145

*method*), 129

InjectNmiBaremetalNode (*class in ironic-client.osc.v1.baremetal\_node*), 103

InspectBaremetalNode (*class in ironic-client.osc.v1.baremetal\_node*), 103

InternalServerError, 82

InvalidAttribute, 145

ironicclient module, 147

ironicclient.client module, 144

ironicclient.common module, 93

ironicclient.common.apiclient module, 88

ironicclient.common.apiclient.base module, 74

ironicclient.common.apiclient.exceptions module, 76

ironicclient.common.base module, 88

ironicclient.common.filecache module, 89

ironicclient.common.http module, 89

ironicclient.common.i18n module, 90

ironicclient.common.utils module, 90

ironicclient.exc module, 145

ironicclient.osc module, 117

ironicclient.osc.plugin module, 117

ironicclient.osc.v1 module, 117

ironicclient.osc.v1.baremetal\_allocation module, 93

ironicclient.osc.v1.baremetal\_chassis module, 95

ironicclient.osc.v1.baremetal\_conductor module, 96

ironicclient.osc.v1.baremetal\_create module, 97

ironicclient.osc.v1.baremetal\_deploy\_template module, 97

ironicclient.osc.v1.baremetal\_driver module, 99

ironicclient.osc.v1.baremetal\_node module, 100

ironicclient.osc.v1.baremetal\_port module, 110

ironicclient.osc.v1.baremetal\_portgroup module, 112

ironicclient.osc.v1.baremetal\_volume\_connection module, 114

ironicclient.osc.v1.baremetal\_volume\_target module, 115

ironicclient.shell module, 146

ironicclient.v1 module, 144

ironicclient.v1.allocation module, 117



- `client.osc.v1.baremetal_allocation`), 93
- `ListBaremetalChassis` (class in `ironic-client.osc.v1.baremetal_chassis`), 95
- `ListBaremetalConductor` (class in `ironic-client.osc.v1.baremetal_conductor`), 96
- `ListBaremetalDeployTemplate` (class in `ironic-client.osc.v1.baremetal_deploy_template`), 98
- `ListBaremetalDriver` (class in `ironic-client.osc.v1.baremetal_driver`), 99
- `ListBaremetalDriverProperty` (class in `ironicclient.osc.v1.baremetal_driver`), 99
- `ListBaremetalDriverRaidProperty` (class in `ironic-client.osc.v1.baremetal_driver`), 99
- `ListBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 104
- `ListBaremetalPort` (class in `ironic-client.osc.v1.baremetal_port`), 111
- `ListBaremetalPortGroup` (class in `ironic-client.osc.v1.baremetal_portgroup`), 112
- `ListBaremetalVolumeConnector` (class in `ironic-client.osc.v1.baremetal_volume_connector`), 114
- `ListBaremetalVolumeTarget` (class in `ironic-client.osc.v1.baremetal_volume_target`), 116
- `ListBIOSSettingBaremetalNode` (class in `ironicclient.osc.v1.baremetal_node`), 104
- `ListTraitsBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 104
- `load_commands()` (`ironic-client.shell.CommandManager` method), 146
- `load_from_file()` (in module `ironic-client.v1.create_resources`), 125
- `log(ironicclient.osc.v1.baremetal_allocation.CreateBaremetalAllocationAttribute)`, 93
- `log(ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocationAttribute)`, 93
- `log(ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocationAttribute)`, 94
- `log(ironicclient.osc.v1.baremetal_allocation.SetBaremetalAllocationAttribute)`, 94
- `log(ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocationAttribute)`, 94
- `log(ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocationAttribute)`, 94
- `log(ironicclient.osc.v1.baremetal_chassis.CreateBaremetalChassisAttribute)`, 95
- `log(ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassisAttribute)`, 95
- `log(ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassisAttribute)`, 95
- `log(ironicclient.osc.v1.baremetal_chassis.SetBaremetalChassisAttribute)`, 95
- `log(ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassisAttribute)`, 96
- `log(ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassisAttribute)`, 96
- `log(ironicclient.osc.v1.baremetal_conductor.ListBaremetalConductorAttribute)`, 96
- `log(ironicclient.osc.v1.baremetal_conductor.ShowBaremetalConductorAttribute)`, 97
- `log(ironicclient.osc.v1.baremetal_create.CreateBaremetalAttribute)`, 97
- `log(ironicclient.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplateAttribute)`, 97
- `log(ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplateAttribute)`, 97
- `log(ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplateAttribute)`, 98
- `log(ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplateAttribute)`, 98
- `log(ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplateAttribute)`, 98
- `log(ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplateAttribute)`, 98
- `log(ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverAttribute)`, 99
- `log(ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverPropertyAttribute)`, 99
- `log(ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverRaidAttribute)`, 99
- `log(ironicclient.osc.v1.baremetal_driver.PassthruCallBaremetalDriverAttribute)`, 99
- `log(ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriverAttribute)`, 99
- `log(ironicclient.osc.v1.baremetal_driver.ShowBaremetalDriverAttribute)`, 100
- `log(ironicclient.osc.v1.baremetal_node.AbortBaremetalNodeAttribute)`, 100
- `log(ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNodeAttribute)`, 100
- `log(ironicclient.osc.v1.baremetal_node.AdoptBaremetalNodeAttribute)`, 101
- `log(ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNodeAttribute)`, 101

- log (ironicclient.osc.v1.baremetal\_node.BootdeviceSetBaremetalNode attribute), 101
- log (ironicclient.osc.v1.baremetal\_node.BootdeviceShowBaremetalNode attribute), 101
- log (ironicclient.osc.v1.baremetal\_node.CleanBaremetalNode attribute), 102
- log (ironicclient.osc.v1.baremetal\_node.ConsoleDisableBaremetalNode attribute), 102
- log (ironicclient.osc.v1.baremetal\_node.ConsoleEnableBaremetalNode attribute), 102
- log (ironicclient.osc.v1.baremetal\_node.ConsoleShowBaremetalNode attribute), 102
- log (ironicclient.osc.v1.baremetal\_node.CreateBaremetalNode attribute), 103
- log (ironicclient.osc.v1.baremetal\_node.DeleteBaremetalNode attribute), 103
- log (ironicclient.osc.v1.baremetal\_node.DeployBaremetalNode attribute), 103
- log (ironicclient.osc.v1.baremetal\_node.InjectNmiBaremetalNode attribute), 103
- log (ironicclient.osc.v1.baremetal\_node.InspectBaremetalNode attribute), 104
- log (ironicclient.osc.v1.baremetal\_node.ListBaremetalNode attribute), 104
- log (ironicclient.osc.v1.baremetal\_node.ListBIOSSettingBaremetalNode attribute), 104
- log (ironicclient.osc.v1.baremetal\_node.ListTraitsBaremetalNode attribute), 104
- log (ironicclient.osc.v1.baremetal\_node.MaintenanceSetBaremetalNode attribute), 105
- log (ironicclient.osc.v1.baremetal\_node.MaintenanceUnsetBaremetalNode attribute), 105
- log (ironicclient.osc.v1.baremetal\_node.ManageBaremetalNode attribute), 105
- log (ironicclient.osc.v1.baremetal\_node.PassthruCallBaremetalNode attribute), 105
- log (ironicclient.osc.v1.baremetal\_node.PassthruListBaremetalNode attribute), 106
- log (ironicclient.osc.v1.baremetal\_node.PowerBaremetalNode attribute), 106
- log (ironicclient.osc.v1.baremetal\_node.PowerOffBaremetalNode attribute), 106
- log (ironicclient.osc.v1.baremetal\_node.PowerOnBaremetalNode attribute), 106
- log (ironicclient.osc.v1.baremetal\_node.ProvideBaremetalNode attribute), 106
- log (ironicclient.osc.v1.baremetal\_node.ProvisionStateBaremetalNode attribute), 107
- log (ironicclient.osc.v1.baremetal\_node.ProvisionStateWithWait attribute), 107
- log (ironicclient.osc.v1.baremetal\_node.RebootBaremetalNode attribute), 107
- log (ironicclient.osc.v1.baremetal\_node.RebuildBaremetalNode attribute), 107
- log (ironicclient.osc.v1.baremetal\_node.RemoveTraitBaremetalNode attribute), 108
- log (ironicclient.osc.v1.baremetal\_node.RescueBaremetalNode attribute), 108
- log (ironicclient.osc.v1.baremetal\_node.SetBaremetalNode attribute), 108
- log (ironicclient.osc.v1.baremetal\_node.ShowBaremetalNode attribute), 108
- log (ironicclient.osc.v1.baremetal\_node.UndeployBaremetalNode attribute), 109
- log (ironicclient.osc.v1.baremetal\_node.UnrescueBaremetalNode attribute), 109
- log (ironicclient.osc.v1.baremetal\_node.UnsetBaremetalNode attribute), 109
- log (ironicclient.osc.v1.baremetal\_node.ValidateBaremetalNode attribute), 109
- log (ironicclient.osc.v1.baremetal\_node.VifAttachBaremetalNode attribute), 110
- log (ironicclient.osc.v1.baremetal\_node.VifDetachBaremetalNode attribute), 110
- log (ironicclient.osc.v1.baremetal\_node.VifListBaremetalNode attribute), 110
- log (ironicclient.osc.v1.baremetal\_port.CreateBaremetalPort attribute), 110
- log (ironicclient.osc.v1.baremetal\_port.DeleteBaremetalPort attribute), 111
- log (ironicclient.osc.v1.baremetal\_port.ListBaremetalPort attribute), 111
- log (ironicclient.osc.v1.baremetal\_port.SetBaremetalPort attribute), 111
- log (ironicclient.osc.v1.baremetal\_port.ShowBaremetalPort attribute), 111
- log (ironicclient.osc.v1.baremetal\_port.UnsetBaremetalPort attribute), 112
- log (ironicclient.osc.v1.baremetal\_portgroup.CreateBaremetalPortGroup attribute), 112
- log (ironicclient.osc.v1.baremetal\_portgroup.DeleteBaremetalPortGroup attribute), 112
- log (ironicclient.osc.v1.baremetal\_portgroup.ListBaremetalPortGroup attribute), 112
- log (ironicclient.osc.v1.baremetal\_portgroup.SetBaremetalPortGroup attribute), 113
- log (ironicclient.osc.v1.baremetal\_portgroup.ShowBaremetalPortGroup attribute), 113
- log (ironicclient.osc.v1.baremetal\_portgroup.UnsetBaremetalPortGroup attribute), 113
- log (ironicclient.osc.v1.baremetal\_volume\_connector.CreateBaremetalVolumeConnector attribute), 114



log( <i>ironicclient.osc.v1.baremetal_volume_connector.DeleteBaremetalVolumeConnector</i> (ironic-client.common.apiclient.exceptions.ExpectationFailed attribute), 114	
log( <i>ironicclient.osc.v1.baremetal_volume_connector.ListBaremetalVolumeConnector</i> (ironic-client.common.apiclient.exceptions.Forbidden attribute), 114	message
log( <i>ironicclient.osc.v1.baremetal_volume_connector.SetBaremetalVolumeConnector</i> (ironic-client.common.apiclient.exceptions.GatewayTimeout attribute), 114	message
log( <i>ironicclient.osc.v1.baremetal_volume_connector.ShowBaremetalVolumeConnector</i> (ironic-client.common.apiclient.exceptions.Gone attribute), 115	message
log( <i>ironicclient.osc.v1.baremetal_volume_connector.UnsetBaremetalVolumeConnector</i> (ironic-client.common.apiclient.exceptions.HTTPClientError attribute), 115	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HTTPClientError attribute), 116	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HTTPClientError attribute), 116	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HttpError attribute), 116	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HttpNotImplemented attribute), 116	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HTTPRedirection attribute), 117	message
log( <i>ironicclient.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget</i> (ironic-client.common.apiclient.exceptions.HttpServerError attribute), 117	message
<b>M</b>	
main() (in module <i>ironicclient.shell</i> ), 146	message
MaintenanceSetBaremetalNode (class in <i>ironicclient.osc.v1.baremetal_node</i> ), 104	message
MaintenanceUnsetBaremetalNode (class in <i>ironicclient.osc.v1.baremetal_node</i> ), 105	message
make_client() (in module <i>ironicclient.osc.plugin</i> ), 117	message
make_configdrive() (in module <i>ironicclient.common.utils</i> ), 92	message
ManageBaremetalNode (class in <i>ironicclient.osc.v1.baremetal_node</i> ), 105	message
Manager (class in <i>ironicclient.common.base</i> ), 88	message
manager_class (ironic-client.common.apiclient.base.Extension attribute), 75	message
ManagerWithFind (class in <i>ironicclient.common.apiclient.base</i> ), 75	message
message (ironic-client.common.apiclient.exceptions.BadGateway attribute), 77	message
message (ironic-client.common.apiclient.exceptions.BadRequest attribute), 77	message
message (ironic-client.common.apiclient.exceptions.Conflict attribute), 78	message

message (ironic-ironicclient.common.utils, 90  
*client.common.apiclient.exceptions.PreconditionFailed*  
 attribute), 84 ironicclient.exc, 145  
 ironicclient.osc, 117

message (ironic-ironicclient.osc.plugin, 117  
*client.common.apiclient.exceptions.ProxyAuthenticationRequired*  
 attribute), 85 ironicclient.osc.v1, 117  
 ironicclient.osc.v1.baremetal\_allocation,

message (ironic-93  
*client.common.apiclient.exceptions.RequestedRangeNotSatisfiable*  
 attribute), 86 ironicclient.osc.v1.baremetal\_chassis,  
 95

message (ironic-ironicclient.osc.v1.baremetal\_conductor,  
*client.common.apiclient.exceptions.RequestEntityTooLarge*  
 attribute), 85 ironicclient.osc.v1.baremetal\_create,

message (ironic-97  
*client.common.apiclient.exceptions.RequestTimeout*  
 attribute), 85 ironicclient.osc.v1.baremetal\_deploy\_template,  
 97

message (ironic-ironicclient.osc.v1.baremetal\_driver,  
*client.common.apiclient.exceptions.RequestUriTooLong*  
 attribute), 86 ironicclient.osc.v1.baremetal\_node,

message (ironic-100  
*client.common.apiclient.exceptions.ServiceUnavailable*  
 attribute), 86 ironicclient.osc.v1.baremetal\_port,  
 110

message (ironic-ironicclient.osc.v1.baremetal\_portgroup,  
*client.common.apiclient.exceptions.Unauthorized*  
 attribute), 87 112  
 ironicclient.osc.v1.baremetal\_volume\_connector,

message (ironic-114  
*client.common.apiclient.exceptions.UnprocessableEntity*  
 attribute), 87 ironicclient.osc.v1.baremetal\_volume\_target,  
 115

message (ironic-ironicclient.shell, 146  
*client.common.apiclient.exceptions.UnsupportedMediaType*  
 attribute), 87 ironicclient.v1, 144  
 ironicclient.v1.allocation, 117

method ironicclient.v1.chassis, 120  
 openstack-baremetal-driver-passthru-carbonade, 122  
 command line option, 18 ironicclient.v1.conductor, 122  
 openstack-baremetal-node-passthru-carbonade, 123  
 command line option, 35

MethodNotAllowed, 82 ironicclient.v1.deploy\_template,  
 126

module ironicclient, 147  
 ironicclient.client, 144  
 ironicclient.common, 93  
 ironicclient.common.apiclient, 88  
 ironicclient.common.apiclient.base, 74  
 ironicclient.common.apiclient.exceptions, 76  
 ironicclient.common.base, 88  
 ironicclient.common.filecache, 89  
 ironicclient.common.http, 89  
 ironicclient.common.il8n, 90

ironicclient.v1.driver, 127  
 ironicclient.v1.events, 128  
 ironicclient.v1.node, 128  
 ironicclient.v1.port, 138  
 ironicclient.v1.portgroup, 139  
 ironicclient.v1.resource\_fields,  
 142  
 ironicclient.v1.utils, 142  
 ironicclient.v1.volume\_connector,  
 142  
 ironicclient.v1.volume\_target,  
 143  
 MultipleChoices, 83

## N

- NAME\_ATTR (*ironic-client.common.apiclient.base.Resource attribute*), 76
- negotiate\_api\_version() (*ironic-client.v1.client.Client method*), 122
- negotiate\_version() (*ironic-client.common.http.VersionNegotiationMixin method*), 89
- node
  - openstack-baremetal-node-abort command line option, 22
  - openstack-baremetal-node-add-trait command line option, 23
  - openstack-baremetal-node-adopt command line option, 23
  - openstack-baremetal-node-bios-setting-list command line option, 24
  - openstack-baremetal-node-bios-setting-show command line option, 25
  - openstack-baremetal-node-boot-device-set command line option, 25
  - openstack-baremetal-node-boot-device-show command line option, 26
  - openstack-baremetal-node-clean command line option, 26
  - openstack-baremetal-node-console-disable command line option, 26
  - openstack-baremetal-node-console-enable command line option, 27
  - openstack-baremetal-node-console-show command line option, 27
  - openstack-baremetal-node-delete command line option, 30
  - openstack-baremetal-node-deploy command line option, 31
  - openstack-baremetal-node-inject-~~file~~ command line option, 31
  - openstack-baremetal-node-inspect command line option, 31
  - openstack-baremetal-node-maintenance-set command line option, 34
  - openstack-baremetal-node-maintenance-unset command line option, 34
  - openstack-baremetal-node-manage command line option, 34
  - openstack-baremetal-node-passthru-call command line option, 35
  - openstack-baremetal-node-passthru-list command line option, 36
  - openstack-baremetal-node-power-off command line option, 36
  - openstack-baremetal-node-power-on command line option, 36
  - openstack-baremetal-node-provide command line option, 37
  - openstack-baremetal-node-reboot command line option, 37
  - openstack-baremetal-node-rebuild command line option, 37
  - openstack-baremetal-node-remove-trait command line option, 38
  - openstack-baremetal-node-rescue command line option, 38
  - openstack-baremetal-node-set command line option, 42
  - openstack-baremetal-node-show command line option, 42
  - openstack-baremetal-node-trait-list command line option, 43
  - openstack-baremetal-node-undeploy command line option, 44
  - openstack-baremetal-node-unrescue command line option, 44
  - openstack-baremetal-node-unset command line option, 46
  - openstack-baremetal-node-validate command line option, 47
  - openstack-baremetal-node-vif-attach command line option, 48
  - openstack-baremetal-node-vif-detach command line option, 48
  - openstack-baremetal-node-vif-list command line option, 49
- Node (*class in ironicclient.v1.node*), 128
- NodeManager (*class in ironicclient.v1.node*), 128
- NotAcceptable, 83
- NotFound, 83
- NoUniqueMatch, 83
- O
  - openstack-baremetal-allocation-create command line option
    - candidate-node <CANDIDATE\_NODES>, 9
    - column COLUMN, 8
    - extra <key=value>, 9
    - fit-width, 8
    - format <FORMATTER>, 8
    - max-width <integer>, 8
    - name <NAME>, 9



```

--node <NODE>, 9
--noindent, 8
--owner <OWNER>, 9
--prefix <PREFIX>, 8
--print-empty, 8
--resource-class
    <RESOURCE_CLASS>, 8
--trait <TRAITS>, 9
--uuid <UUID>, 9
--wait <time-out>, 9
-c COLUMN, 8
-f <FORMATTER>, 8
openstack-baremetal-allocation-delete
    command line option
    allocation, 9
openstack-baremetal-allocation-list
    command line option
--column COLUMN, 10
--fields <field>, 11
--fit-width, 10
--format <FORMATTER>, 10
--limit <limit>, 10
--long, 10
--marker <allocation>, 10
--max-width <integer>, 10
--node <node>, 10
--noindent, 10
--owner <owner>, 10
--print-empty, 10
--quote <QUOTE_MODE>, 10
--resource-class
    <resource_class>, 10
--sort <key>[:<direction>], 10
--sort-column SORT_COLUMN, 10
--state <state>, 10
-c COLUMN, 10
-f <FORMATTER>, 10
openstack-baremetal-allocation-set
    command line option
--extra <key=value>, 11
--name <name>, 11
    allocation, 11
openstack-baremetal-allocation-show
    command line option
--column COLUMN, 11
--fields <field>, 12
--fit-width, 12
--format <FORMATTER>, 11
--max-width <integer>, 11
--noindent, 11
--prefix <PREFIX>, 11
--print-empty, 12
-c COLUMN, 11
-f <FORMATTER>, 11
    id, 12
openstack-baremetal-allocation-unset
    command line option
--extra <key>, 12
--name, 12
    allocation, 12
openstack-baremetal-chassis-create
    command line option
--column COLUMN, 13
--description <description>, 13
--extra <key=value>, 13
--fit-width, 13
--format <FORMATTER>, 13
--max-width <integer>, 13
--noindent, 13
--prefix <PREFIX>, 13
--print-empty, 13
--uuid <uuid>, 13
-c COLUMN, 13
-f <FORMATTER>, 13
openstack-baremetal-chassis-delete
    command line option
    chassis, 13
openstack-baremetal-chassis-list
    command line option
--column COLUMN, 14
--fields <field>, 14
--fit-width, 14
--format <FORMATTER>, 14
--limit <limit>, 14
--long, 14
--marker <chassis>, 14
--max-width <integer>, 14
--noindent, 14
--print-empty, 14
--quote <QUOTE_MODE>, 14
--sort <key>[:<direction>], 15
--sort-column SORT_COLUMN, 14
-c COLUMN, 14
-f <FORMATTER>, 14
openstack-baremetal-chassis-set
    command line option
--description <description>, 15
--extra <key=value>, 15
    chassis, 15
openstack-baremetal-chassis-show
    command line option
--column COLUMN, 15

```

```

--fields <field>, 16
--fit-width, 16
--format <FORMATTER>, 15
--max-width <integer>, 16
--noindent, 15
--prefix <PREFIX>, 15
--print-empty, 16
-c COLUMN, 15
-f <FORMATTER>, 15
chassis, 16
openstack-baremetal-chassis-unset
    command line option
    --description, 16
    --extra <key>, 16
    chassis, 16
openstack-baremetal-create
    command line option
    file, 16
openstack-baremetal-driver-list
    command line option
    --column COLUMN, 17
    --fit-width, 17
    --format <FORMATTER>, 17
    --long, 17
    --max-width <integer>, 17
    --noindent, 17
    --print-empty, 17
    --quote <QUOTE_MODE>, 17
    --sort-column SORT_COLUMN, 17
    --type <type>, 17
    -c COLUMN, 17
    -f <FORMATTER>, 17
openstack-baremetal-driver-passthru-callb
    command line option
    --arg <key=value>, 18
    --column COLUMN, 18
    --fit-width, 18
    --format <FORMATTER>, 18
    --http-method <http-method>, 18
    --max-width <integer>, 18
    --noindent, 18
    --prefix <PREFIX>, 18
    --print-empty, 18
    -c COLUMN, 18
    -f <FORMATTER>, 18
    driver, 18
    method, 18
openstack-baremetal-driver-passthru-opts
    command line option
    --column COLUMN, 19
    --fit-width, 19
--format <FORMATTER>, 19
--max-width <integer>, 19
--noindent, 19
--print-empty, 19
--quote <QUOTE_MODE>, 19
--sort-column SORT_COLUMN, 19
-c COLUMN, 19
-f <FORMATTER>, 19
driver, 19
openstack-baremetal-driver-property-list
    command line option
    --column COLUMN, 20
    --fit-width, 20
    --format <FORMATTER>, 20
    --max-width <integer>, 20
    --noindent, 20
    --print-empty, 20
    --quote <QUOTE_MODE>, 20
    --sort-column SORT_COLUMN, 20
    -c COLUMN, 20
    -f <FORMATTER>, 20
    driver, 20
openstack-baremetal-driver-raid-property-list
    command line option
    --column COLUMN, 21
    --fit-width, 21
    --format <FORMATTER>, 21
    --max-width <integer>, 21
    --noindent, 21
    --print-empty, 21
    --quote <QUOTE_MODE>, 21
    --sort-column SORT_COLUMN, 21
    -c COLUMN, 21
    -f <FORMATTER>, 21
    driver, 21
openstack-baremetal-driver-show
    command line option
    --column COLUMN, 22
    --fit-width, 22
    --format <FORMATTER>, 22
    --max-width <integer>, 22
    --noindent, 22
    --prefix <PREFIX>, 22
    --print-empty, 22
    -c COLUMN, 22
    -f <FORMATTER>, 22
    driver, 22
openstack-baremetal-node-abort
    command line option
    node, 22
openstack-baremetal-node-add-trait

```

command line option  
 node, 23  
 trait, 23  
 openstack-baremetal-node-adopt  
     command line option  
     --wait <time-out>, 23  
     node, 23  
 openstack-baremetal-node-bios-setting-hosts  
     command line option  
     --column COLUMN, 23  
     --fit-width, 24  
     --format <FORMATTER>, 23  
     --max-width <integer>, 23  
     --noindent, 23  
     --print-empty, 24  
     --quote <QUOTE\_MODE>, 23  
     --sort-column SORT\_COLUMN, 24  
     -c COLUMN, 23  
     -f <FORMATTER>, 23  
     node, 24  
 openstack-baremetal-node-bios-setting-show  
     command line option  
     --column COLUMN, 24  
     --fit-width, 24  
     --format <FORMATTER>, 24  
     --max-width <integer>, 24  
     --noindent, 24  
     --prefix <PREFIX>, 24  
     --print-empty, 24  
     -c COLUMN, 24  
     -f <FORMATTER>, 24  
     node, 25  
     setting name, 25  
 openstack-baremetal-node-boot-device-set  
     command line option  
     --persistent, 25  
     device, 25  
     node, 25  
 openstack-baremetal-node-boot-device-show  
     command line option  
     --column COLUMN, 25  
     --fit-width, 26  
     --format <FORMATTER>, 25  
     --max-width <integer>, 25  
     --noindent, 25  
     --prefix <PREFIX>, 25  
     --print-empty, 26  
     --supported, 26  
     -c COLUMN, 25  
     -f <FORMATTER>, 25  
     node, 26  
     openstack-baremetal-node-clean  
         command line option  
         --clean-steps <clean-steps>, 26  
         --wait <time-out>, 26  
         node, 26  
     openstack-baremetal-node-console-disable  
         command line option  
         node, 26  
     openstack-baremetal-node-console-enable  
         command line option  
         node, 27  
     openstack-baremetal-node-console-show  
         command line option  
         --column COLUMN, 27  
         --fit-width, 27  
         --format <FORMATTER>, 27  
         --max-width <integer>, 27  
         --noindent, 27  
         --prefix <PREFIX>, 27  
         --print-empty, 27  
         node, 27  
     openstack-baremetal-node-create  
         command line option  
         --automated-clean, 30  
         --bios-interface  
             <bios\_interface>, 29  
         --boot-interface  
             <boot\_interface>, 29  
         --chassis-uuid <chassis>, 29  
         --column COLUMN, 28  
         --conductor-group  
             <conductor\_group>, 30  
         --console-interface  
             <console\_interface>, 29  
         --deploy-interface  
             <deploy\_interface>, 29  
         --description <description>, 30  
         --driver <driver>, 29  
         --driver-info <key=value>, 29  
         --extra <key=value>, 29  
         --fit-width, 28  
         --format <FORMATTER>, 28  
         --inspect-interface  
             <inspect\_interface>, 29  
         --lessee <lessee>, 30  
         --management-interface  
             <management\_interface>, 29  
         --max-width <integer>, 28  
         --name <name>, 29

```

--network-data <network data>,
    29
--network-interface
    <network_interface>, 29
--noindent, 28
--owner <owner>, 30
--power-interface
    <power_interface>, 29
--prefix <PREFIX>, 28
--print-empty, 29
--property <key=value>, 29
--raid-interface
    <raid_interface>, 30
--rescue-interface
    <rescue_interface>, 30
--resource-class
    <resource_class>, 30
--storage-interface
    <storage_interface>, 30
--uuid <uuid>, 29
--vendor-interface
    <vendor_interface>, 30
-c COLUMN, 28
-f <FORMATTER>, 28
openstack-baremetal-node-delete
    command line option
    node, 30
openstack-baremetal-node-deploy
    command line option
    --config-drive <config-drive>,
        31
    --wait <time-out>, 31
    node, 31
openstack-baremetal-node-inject-nmi
    command line option
    node, 31
openstack-baremetal-node-inspect
    command line option
    --wait <time-out>, 31
    node, 31
openstack-baremetal-node-list
    command line option
    --associated, 33
    --chassis <chassis UUID>, 33
    --column COLUMN, 32
    --conductor <conductor>, 33
    --conductor-group
        <conductor_group>, 33
    --description-contains
        <description_contains>, 33
    --driver <driver>, 33
    --fault <fault>, 33
    --fields <field>, 34
    --fit-width, 32
    --format <FORMATTER>, 32
    --lessee <lessee>, 33
    --limit <limit>, 32
    --long, 33
    --maintenance, 33
    --marker <node>, 33
    --max-width <integer>, 32
    --no-maintenance, 33
    --no-retired, 33
    --noindent, 32
    --owner <owner>, 33
    --print-empty, 32
    --provision-state <provision
        state>, 33
    --quote <QUOTE_MODE>, 32
    --resource-class <resource
        class>, 33
    --retired, 33
    --sort <key>[:<direction>], 33
    --sort-column SORT_COLUMN, 32
    --unassociated, 33
    -c COLUMN, 32
    -f <FORMATTER>, 32
openstack-baremetal-node-maintenance-set
    command line option
    --reason <reason>, 34
    node, 34
openstack-baremetal-node-maintenance-unset
    command line option
    node, 34
openstack-baremetal-node-manage
    command line option
    --wait <time-out>, 34
    node, 34
openstack-baremetal-node-passthru-call
    command line option
    --arg <key=value>, 35
    --http-method <http-method>, 35
    method, 35
    node, 35
openstack-baremetal-node-passthru-list
    command line option
    --column COLUMN, 35
    --fit-width, 35
    --format <FORMATTER>, 35
    --max-width <integer>, 35
    --noindent, 35
    --print-empty, 36

```

```

--quote <QUOTE_MODE>, 35
--sort-column SORT_COLUMN, 36
-c COLUMN, 35
-f <FORMATTER>, 35
node, 36
openstack-baremetal-node-power-off
    command line option
--power-timeout
    <power-timeout>, 36
--soft, 36
node, 36
openstack-baremetal-node-power-on
    command line option
--power-timeout
    <power-timeout>, 36
node, 36
openstack-baremetal-node-provide
    command line option
--wait <time-out>, 37
node, 37
openstack-baremetal-node-reboot
    command line option
--power-timeout
    <power-timeout>, 37
--soft, 37
node, 37
openstack-baremetal-node-rebuild
    command line option
--config-drive <config-drive>,
    37
--wait <time-out>, 37
node, 37
openstack-baremetal-node-remove-trait
    command line option
--all, 38
node, 38
trait, 38
openstack-baremetal-node-rescue
    command line option
--rescue-password
    <rescue-password>, 38
--wait <time-out>, 38
node, 38
openstack-baremetal-node-set
    command line option
--automated-clean, 41
--bios-interface
    <bios_interface>, 39
--boot-interface
    <boot_interface>, 39
--chassis-uuid <chassis UUID>,
    39
--conductor-group
    <conductor_group>, 41
--console-interface
    <console_interface>, 40
--deploy-interface
    <deploy_interface>, 40
--description <description>, 41
--driver <driver>, 39
--driver-info <key=value>, 41
--extra <key=value>, 41
--inspect-interface
    <inspect_interface>, 40
--instance-info <key=value>, 41
--instance-uuid <uuid>, 39
--lessee <lessee>, 41
--management-interface
    <management_interface>, 40
--name <name>, 39
--network-data <network data>,
    40
--network-interface
    <network_interface>, 40
--owner <owner>, 41
--power-interface
    <power_interface>, 40
--property <key=value>, 41
--protected, 41
--protected-reason
    <protected_reason>, 41
--raid-interface
    <raid_interface>, 40
--rescue-interface
    <rescue_interface>, 40
--reset-bios-interface, 39
--reset-boot-interface, 40
--reset-console-interface, 40
--reset-deploy-interface, 40
--reset-inspect-interface, 40
--reset-interfaces, 41
--reset-management-interface, 40
--reset-network-interface, 40
--reset-power-interface, 40
--reset-raid-interface, 40
--reset-rescue-interface, 40
--reset-storage-interface, 41
--reset-vendor-interface, 41
--resource-class
    <resource_class>, 41
--retired, 41
--retired-reason

```

```

    <retired_reason>, 41
--storage-interface
    <storage_interface>, 40
--target-raid-config
    <target_raid_config>, 41
--vendor-interface
    <vendor_interface>, 41
node, 42
openstack-baremetal-node-show
    command line option
--column COLUMN, 42
--fields <field>, 42
--fit-width, 42
--format <FORMATTER>, 42
--instance, 42
--max-width <integer>, 42
--noindent, 42
--prefix <PREFIX>, 42
--print-empty, 42
-c COLUMN, 42
-f <FORMATTER>, 42
node, 42
openstack-baremetal-node-trait-list
    command line option
--column COLUMN, 43
--fit-width, 43
--format <FORMATTER>, 43
--max-width <integer>, 43
--noindent, 43
--print-empty, 43
--quote <QUOTE_MODE>, 43
--sort-column SORT_COLUMN, 43
-c COLUMN, 43
-f <FORMATTER>, 43
node, 43
openstack-baremetal-node-undeploy
    command line option
--wait <time-out>, 44
node, 44
openstack-baremetal-node-unrescue
    command line option
--wait <time-out>, 44
node, 44
openstack-baremetal-node-unset
    command line option
--automated-clean, 46
--bios-interface, 45
--boot-interface, 45
--chassis-uuid, 45
--conductor-group, 46
--console-interface, 45
--deploy-interface, 45
--description, 46
--driver-info <key>, 45
--extra <key>, 45
--inspect-interface, 45
--instance-info <key>, 45
--instance-uuid, 45
--lessee, 46
--management-interface, 46
--name, 45
--network-data, 45
--network-interface, 46
--owner, 46
--power-interface, 46
--property <key>, 45
--protected, 46
--protected-reason, 46
--raid-interface, 46
--rescue-interface, 46
--resource-class, 45
--retired, 46
--retired-reason, 46
--storage-interface, 46
--target-raid-config, 45
--vendor-interface, 46
node, 46
openstack-baremetal-node-validate
    command line option
--column COLUMN, 47
--fit-width, 47
--format <FORMATTER>, 47
--max-width <integer>, 47
--noindent, 47
--print-empty, 47
--quote <QUOTE_MODE>, 47
--sort-column SORT_COLUMN, 47
-c COLUMN, 47
-f <FORMATTER>, 47
node, 47
openstack-baremetal-node-vif-attach
    command line option
--port-uuid <port-uuid>, 48
--vif-info <key=value>, 48
node, 48
vif-id, 48
openstack-baremetal-node-vif-detach
    command line option
node, 48
vif-id, 48
openstack-baremetal-node-vif-list
    command line option

```

```

--column COLUMN, 49
--fit-width, 49
--format <FORMATTER>, 48
--max-width <integer>, 49
--noindent, 49
--print-empty, 49
--quote <QUOTE_MODE>, 49
--sort-column SORT_COLUMN, 49
-c COLUMN, 49
-f <FORMATTER>, 48
node, 49
openstack-baremetal-port-create
    command line option
--column COLUMN, 50
--extra <key=value>, 50
--fit-width, 50
--format <FORMATTER>, 49
--is-smartnic, 50
--local-link-connection
    <key=value>, 50
--max-width <integer>, 50
--node <uuid>, 50
--noindent, 50
--physical-network <physical
    network>, 50
--port-group <uuid>, 50
--prefix <PREFIX>, 50
--print-empty, 50
--pxe-enabled <boolean>, 50
--uuid <uuid>, 50
-c COLUMN, 50
-f <FORMATTER>, 49
-l <key=value>, 50
address, 50
openstack-baremetal-port-delete
    command line option
port, 51
openstack-baremetal-port-group-create
    command line option
--address <mac-address>, 52
--column COLUMN, 51
--extra <key=value>, 52
--fit-width, 51
--format <FORMATTER>, 51
--max-width <integer>, 51
--mode <MODE>, 52
--name <NAME>, 52
--node <uuid>, 51
--noindent, 51
--prefix <PREFIX>, 51
--print-empty, 51
--property <key=value>, 52
--support-standalone-ports, 52
--unsupport-standalone-ports, 52
--uuid <UUID>, 52
-c COLUMN, 51
-f <FORMATTER>, 51
openstack-baremetal-port-group-delete
    command line option
port group, 52
openstack-baremetal-port-group-list
    command line option
--address <mac-address>, 53
--column COLUMN, 53
--fields <field>, 53
--fit-width, 53
--format <FORMATTER>, 53
--limit <limit>, 53
--long, 53
--marker <port group>, 53
--max-width <integer>, 53
--node <node>, 53
--noindent, 53
--print-empty, 53
--quote <QUOTE_MODE>, 53
--sort <key>[:<direction>], 53
--sort-column SORT_COLUMN, 53
-c COLUMN, 53
-f <FORMATTER>, 53
openstack-baremetal-port-group-set
    command line option
--address <mac-address>, 54
--extra <key=value>, 54
--mode <MODE>, 54
--name <name>, 54
--node <uuid>, 54
--property <key=value>, 54
--support-standalone-ports, 54
--unsupport-standalone-ports, 54
port group, 54
openstack-baremetal-port-group-show
    command line option
--address, 55
--column COLUMN, 55
--fields <field>, 55
--fit-width, 55
--format <FORMATTER>, 55
--max-width <integer>, 55
--noindent, 55
--prefix <PREFIX>, 55
--print-empty, 55
-c COLUMN, 55

```

```

    -f <FORMATTER>, 55
    id, 55
openstack-baremetal-port-group-unset
    command line option
    --address, 55
    --extra <key>, 56
    --name, 55
    --property <key>, 56
    port group, 56
openstack-baremetal-port-list
    command line option
    --address <mac-address>, 57
    --column COLUMN, 56
    --fields <field>, 57
    --fit-width, 56
    --format <FORMATTER>, 56
    --limit <limit>, 57
    --long, 57
    --marker <port>, 57
    --max-width <integer>, 56
    --node <node>, 57
    --noindent, 56
    --port-group <port group>, 57
    --print-empty, 56
    --quote <QUOTE_MODE>, 56
    --sort <key>[:<direction>], 57
    --sort-column SORT_COLUMN, 56
    -c COLUMN, 56
    -f <FORMATTER>, 56
openstack-baremetal-port-set
    command line option
    --address <address>, 57
    --extra <key=value>, 57
    --is-smartnic, 58
    --local-link-connection
        <key=value>, 58
    --node <uuid>, 57
    --physical-network <physical
        network>, 58
    --port-group <uuid>, 57
    --pxe-disabled, 58
    --pxe-enabled, 58
    port, 58
openstack-baremetal-port-show
    command line option
    --address, 59
    --column COLUMN, 58
    --fields <field>, 59
    --fit-width, 58
    --format <FORMATTER>, 58
    --max-width <integer>, 58
    --noindent, 58
    --prefix <PREFIX>, 58
    --print-empty, 59
    -c COLUMN, 58
    -f <FORMATTER>, 58
    id, 59
openstack-baremetal-port-unset
    command line option
    --extra <key>, 59
    --is-smartnic, 59
    --physical-network, 59
    --port-group, 59
    port, 59
openstack-baremetal-volume-connector-create
    command line option
    --column COLUMN, 60
    --connector-id <connector id>,
        60
    --extra <key=value>, 60
    --fit-width, 60
    --format <FORMATTER>, 60
    --max-width <integer>, 60
    --node <uuid>, 60
    --noindent, 60
    --prefix <PREFIX>, 60
    --print-empty, 60
    --type <type>, 60
    --uuid <uuid>, 60
    -c COLUMN, 60
    -f <FORMATTER>, 60
openstack-baremetal-volume-connector-delete
    command line option
    volume connector, 61
openstack-baremetal-volume-connector-list
    command line option
    --column COLUMN, 61
    --fields <field>, 62
    --fit-width, 61
    --format <FORMATTER>, 61
    --limit <limit>, 62
    --long, 62
    --marker <volume connector>, 62
    --max-width <integer>, 61
    --node <node>, 62
    --noindent, 61
    --print-empty, 61
    --quote <QUOTE_MODE>, 61
    --sort <key>[:<direction>], 62
    --sort-column SORT_COLUMN, 61
    -c COLUMN, 61
    -f <FORMATTER>, 61

```



```

openstack-baremetal-volume-connector-set-format <FORMATTER>, 65
    command line option
    --limit <limit>, 66
    --connector-id <connector id>, 62
    --marker <volume target>, 66
    --extra <key=value>, 62
    --max-width <integer>, 66
    --node <node>, 66
    --type <type>, 62
    --noindent, 66
    volume connector, 62
    --print-empty, 66
openstack-baremetal-volume-connector-show
    command line option
    --quote <QUOTE_MODE>, 65
    --sort <key>[:<direction>], 66
    --column COLUMN, 63
    --sort-column SORT_COLUMN, 66
    --fields <field>, 63
    -c COLUMN, 65
    --fit-width, 63
    -f <FORMATTER>, 65
    --format <FORMATTER>, 63
    openstack-baremetal-volume-target-set
    --max-width <integer>, 63
    command line option
    --boot-index <boot index>, 67
    --noindent, 63
    --extra <key=value>, 67
    --node <uuid>, 67
    --prefix <PREFIX>, 63
    --property <key=value>, 67
    --print-empty, 63
    --type <volume type>, 67
    -c COLUMN, 63
    --volume-id <volume id>, 67
    -f <FORMATTER>, 63
    id, 63
    volume target, 67
openstack-baremetal-volume-connector-unset
    command line option
    openstack-baremetal-volume-target-show
    --extra <key>, 63
    command line option
    volume connector, 64
    --column COLUMN, 67
openstack-baremetal-volume-target-create
    command line option
    fields <field>, 68
    --boot-index <boot index>, 65
    --fit-width, 67
    --column COLUMN, 64
    --format <FORMATTER>, 67
    --max-width <integer>, 67
    --extra <key=value>, 65
    --noindent, 67
    --format <FORMATTER>, 64
    --prefix <PREFIX>, 67
    --max-width <integer>, 64
    --print-empty, 68
    --node <uuid>, 64
    -c COLUMN, 67
    --noindent, 64
    -f <FORMATTER>, 67
    id, 68
    --prefix <PREFIX>, 64
    openstack-baremetal-volume-target-unset
    --print-empty, 64
    command line option
    --property <key=value>, 64
    --extra <key>, 68
    --type <volume type>, 64
    --property <key>, 68
    --uuid <uuid>, 65
    volume target, 68
    --volume-id <volume id>, 65
    -c COLUMN, 64
    -f <FORMATTER>, 64
P
PassthruCallBaremetalDriver (class in
    ironclient.osc.v1.baremetal_driver), 99
PassthruCallBaremetalNode (class in
    ironclient.osc.v1.baremetal_node), 105
PassthruListBaremetalDriver (class in
    ironclient.osc.v1.baremetal_driver),
    100
PassthruListBaremetalNode (class in
    ironclient.osc.v1.baremetal_node), 105
openstack-baremetal-volume-target-delete
    command line option
    volume target, 65
openstack-baremetal-volume-target-list
    command line option
    --column COLUMN, 65
    --fields <field>, 66
    --fit-width, 66

```

- PaymentRequired, 84
  - poll() (in module *ironicclient.common.utils*), 92
  - port
    - openstack-baremetal-port-delete
      - command line option, 51
    - openstack-baremetal-port-set
      - command line option, 58
    - openstack-baremetal-port-unset
      - command line option, 59
  - Port (class in *ironicclient.v1.port*), 138
  - port group
    - openstack-baremetal-port-group-delete
      - command line option, 52
    - openstack-baremetal-port-group-set
      - command line option, 54
    - openstack-baremetal-port-group-unset
      - command line option, 56
  - Portgroup (class in *ironicclient.v1.portgroup*), 139
  - PortgroupManager (class in *ironicclient.v1.portgroup*), 139
  - PortManager (class in *ironicclient.v1.port*), 138
  - POWER\_STATE (ironic-client.osc.v1.baremetal\_node.PowerOffBaremetalNode attribute), 106
  - POWER\_STATE (ironic-client.osc.v1.baremetal\_node.PowerOnBaremetalNode attribute), 106
  - PowerBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 106
  - PowerOffBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 106
  - PowerOnBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 106
  - PreconditionFailed, 84
  - properties() (ironic-client.v1.driver.DriverManager method), 127
  - ProvideBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 106
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.AbortBaremetalNode attribute), 100
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.AdoptBaremetalNode attribute), 101
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.CleanBaremetalNode attribute), 102
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.DeployBaremetalNode attribute), 103
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.InspectBaremetalNode attribute), 104
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.ManageBaremetalNode attribute), 105
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.ProvideBaremetalNode attribute), 106
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.RebuildBaremetalNode attribute), 107
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.RescueBaremetalNode attribute), 108
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.UndeployBaremetalNode attribute), 109
  - PROVISION\_STATE (ironic-client.osc.v1.baremetal\_node.UnrescueBaremetalNode attribute), 109
  - PROVISION\_STATES (ironic-client.osc.v1.baremetal\_node.ListBaremetalNode attribute), 104
  - ProvisionStateBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 106
  - ProvisionStateWithWait (class in *ironicclient.osc.v1.baremetal\_node*), 107
  - ProxyAuthenticationRequired, 84
  - put() (*ironicclient.common.apiclient.base.CrudManager* method), 75
- ## R
- raid\_logical\_disk\_properties() (ironicclient.v1.driver.DriverManager method), 127
  - raw\_request() (ironic-client.common.http.SessionClient method), 89
  - RebootBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 107
  - RebuildBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 107
  - Node\_all\_traits() (ironic-client.v1.node.NodeManager method), 133
  - Node\_trait() (ironic-client.v1.node.NodeManager method), 133

RemoveTraitBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 108

ReplaceLatestVersion (class in *ironicclient.osc.plugin*), 117

RequestedRangeNotSatisfiable, 86

RequestEntityTooLarge, 85

RequestTimeout, 85

RequestUriTooLong, 85

RescueBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 108

Resource (class in *ironicclient.common.apiclient.base*), 76

Resource (class in *ironicclient.common.base*), 89

Resource (class in *ironicclient.v1.resource\_fields*), 142

resource\_class (ironicclient.common.apiclient.base.BaseManager attribute), 74

resource\_class (ironicclient.v1.allocation.AllocationManager attribute), 119

resource\_class (ironicclient.v1.chassis.ChassisManager attribute), 121

resource\_class (ironicclient.v1.conductor.ConductorManager attribute), 123

resource\_class (ironicclient.v1.deploy\_template.DeployTemplateManager attribute), 126

resource\_class (ironicclient.v1.driver.DriverManager attribute), 127

resource\_class (ironicclient.v1.events.EventManager attribute), 128

resource\_class (ironicclient.v1.node.NodeManager attribute), 133

resource\_class (ironicclient.v1.port.PortManager attribute), 139

resource\_class (ironicclient.v1.portgroup.PortgroupManager attribute), 141

resource\_class (ironicclient.v1.volume\_connector.VolumeConnectorManager attribute), 143

resource\_class (ironicclient.v1.volume\_target.VolumeTargetManager attribute), 144

resource\_class() (ironicclient.common.base.Manager property), 88

retrieve\_data() (in module *ironicclient.common.filecache*), 89

run\_hooks() (ironicclient.common.apiclient.base.HookableMixin class method), 75

## S

save\_data() (in module *ironicclient.common.filecache*), 89

ServiceUnavailable, 86

SessionClient (class in *ironicclient.common.http*), 89

set\_boot\_device() (ironicclient.v1.node.NodeManager method), 133

set\_console\_mode() (ironicclient.v1.node.NodeManager method), 133

set\_loaded() (ironicclient.common.apiclient.base.Resource method), 76

set\_maintenance() (ironicclient.v1.node.NodeManager method), 134

set\_power\_state() (ironicclient.v1.node.NodeManager method), 134

set\_provision\_state() (ironicclient.v1.node.NodeManager method), 134

set\_target\_raid\_config() (ironicclient.v1.node.NodeManager method), 135

set\_traits() (ironicclient.v1.node.NodeManager method), 135

SetBaremetalAllocation (class in *ironicclient.osc.v1.baremetal\_allocation*), 94

SetBaremetalChassis (class in *ironicclient.osc.v1.baremetal\_chassis*), 95

SetBaremetalDeployTemplate (class in *ironicclient.osc.v1.baremetal\_deploy\_template*), 98

SetBaremetalNode (class in *ironicclient.osc.v1.baremetal\_node*), 108

SetBaremetalPort (class in *ironic-*

*client.osc.v1.baremetal\_port*), 111

SetBaremetalPortGroup (class in *ironic-client.osc.v1.baremetal\_portgroup*), 113

SetBaremetalVolumeConnector (class in *ironic-client.osc.v1.baremetal\_volume\_connector*), 114

SetBaremetalVolumeTarget (class in *ironic-client.osc.v1.baremetal\_volume\_target*), 116

setting name

  openstack-baremetal-node-bios-setup (command line option), 25

ShowBaremetalAllocation (class in *ironic-client.osc.v1.baremetal\_allocation*), 94

ShowBaremetalChassis (class in *ironic-client.osc.v1.baremetal\_chassis*), 96

ShowBaremetalConductor (class in *ironic-client.osc.v1.baremetal\_conductor*), 96

ShowBaremetalDeployTemplate (class in *ironic-client.osc.v1.baremetal\_deploy\_template*), 98

ShowBaremetalDriver (class in *ironic-client.osc.v1.baremetal\_driver*), 100

ShowBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 108

ShowBaremetalPort (class in *ironic-client.osc.v1.baremetal\_port*), 111

ShowBaremetalPortGroup (class in *ironic-client.osc.v1.baremetal\_portgroup*), 113

ShowBaremetalVolumeConnector (class in *ironic-client.osc.v1.baremetal\_volume\_connector*), 115

ShowBaremetalVolumeTarget (class in *ironic-client.osc.v1.baremetal\_volume\_target*), 116

sort\_fields() (*ironic-client.v1.resource\_fields.Resource* property), 142

sort\_labels() (*ironic-client.v1.resource\_fields.Resource* property), 142

split\_and\_deserialize() (in module *ironicclient.common.utils*), 93

start\_section() (*ironic-client.common.utils.HelpFormatter* method), 90

states() (*ironicclient.v1.node.NodeManager* method), 136

StateTransitionFailed, 145

StateTransitionTimeout, 146

SUPPORTED\_HOOKS (*ironic-client.common.apiclient.base.Extension* attribute), 75

## T

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.CreateBaremetalAllocation* method), 93

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.DeleteBaremetalAllocation* method), 93

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.ListBaremetalAllocations* method), 94

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.SetBaremetalAllocation* method), 94

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.ShowBaremetalAllocation* method), 94

take\_action() (*ironic-client.osc.v1.baremetal\_allocation.UnsetBaremetalAllocation* method), 94

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.CreateBaremetalChassis* method), 95

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.DeleteBaremetalChassis* method), 95

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.ListBaremetalChassis* method), 95

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.SetBaremetalChassis* method), 95

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.ShowBaremetalChassis* method), 96

take\_action() (*ironic-client.osc.v1.baremetal\_chassis.UnsetBaremetalChassis* method), 96

take\_action() (*ironic-client.osc.v1.baremetal\_conductor.ListBaremetalConductors* method), 96

take\_action() (*ironic-client.osc.v1.baremetal\_conductor.ShowBaremetalConductor* method), 97

take_action()	(ironic- client.osc.v1.baremetal_create.CreateBaremetal method), 97	(ironic- client.osc.v1.baremetal_node.ConsoleDisableBaremetalNode method), 102
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplate method), 97	(ironic- client.osc.v1.baremetal_node.ConsoleEnableBaremetalNode method), 102
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate method), 97	(ironic- client.osc.v1.baremetal_node.ConsoleShowBaremetalNode method), 102
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplates method), 98	(ironic- client.osc.v1.baremetal_node.CreateBaremetalNode method), 103
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate method), 98	(ironic- client.osc.v1.baremetal_node.DeleteBaremetalNode method), 103
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate method), 98	(ironic- client.osc.v1.baremetal_node.InjectNmiBaremetalNode method), 103
take_action()	(ironic- client.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate method), 98	(ironic- client.osc.v1.baremetal_node.ListBaremetalNode method), 104
take_action()	(ironic- client.osc.v1.baremetal_driver.ListBaremetalDriver method), 99	(ironic- client.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode method), 104
take_action()	(ironic- client.osc.v1.baremetal_driver.ListBaremetalDriverPart method), 99	(ironic- client.osc.v1.baremetal_node.ListTraitsBaremetalNode method), 104
take_action()	(ironic- client.osc.v1.baremetal_driver.ListBaremetalDriverProperty method), 99	(ironic- client.osc.v1.baremetal_node.MaintenanceSetBaremetalNode method), 105
take_action()	(ironic- client.osc.v1.baremetal_driver.PassthruCallBaremetalDriver method), 99	(ironic- client.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode method), 105
take_action()	(ironic- client.osc.v1.baremetal_driver.PassthruListBaremetalDriver method), 100	(ironic- client.osc.v1.baremetal_node.PassthruCallBaremetalNode method), 105
take_action()	(ironic- client.osc.v1.baremetal_driver.ShowBaremetalDriver method), 100	(ironic- client.osc.v1.baremetal_node.PassthruListBaremetalNode method), 106
take_action()	(ironic- client.osc.v1.baremetal_node.AddTraitBaremetalNode method), 100	(ironic- client.osc.v1.baremetal_node.PowerBaremetalNode method), 106
take_action()	(ironic- client.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode method), 101	(ironic- client.osc.v1.baremetal_node.ProvisionStateBaremetalNode method), 107
take_action()	(ironic- client.osc.v1.baremetal_node.BootdeviceSetBaremetalNode method), 101	(ironic- client.osc.v1.baremetal_node.ProvisionStateWithWait method), 107
take_action()	(ironic- client.osc.v1.baremetal_node.BootdeviceShowBaremetalNode method), 101	(ironic- client.osc.v1.baremetal_node.RebootBaremetalNode method), 107



take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</i>	<i>method</i> ), 108	<i>client.osc.v1.baremetal_portgroup.SetBaremetalPortGroup</i>	<i>method</i> ), 113
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.SetBaremetalNode</i>	<i>method</i> ), 108	<i>client.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup</i>	<i>method</i> ), 113
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.ShowBaremetalNode</i>	<i>method</i> ), 108	<i>client.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup</i>	<i>method</i> ), 113
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.UnsetBaremetalNode</i>	<i>method</i> ), 109	<i>client.osc.v1.baremetal_volume_connector.CreateBaremetalVolume</i>	<i>method</i> ), 114
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.ValidateBaremetalNode</i>	<i>method</i> ), 109	<i>client.osc.v1.baremetal_volume_connector.DeleteBaremetalVolume</i>	<i>method</i> ), 114
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.VifAttachBaremetalNode</i>	<i>method</i> ), 110	<i>client.osc.v1.baremetal_volume_connector.ListBaremetalVolumes</i>	<i>method</i> ), 114
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.VifDetachBaremetalNode</i>	<i>method</i> ), 110	<i>client.osc.v1.baremetal_volume_connector.SetBaremetalVolume</i>	<i>method</i> ), 114
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_node.VifListBaremetalNode</i>	<i>method</i> ), 110	<i>client.osc.v1.baremetal_volume_connector.ShowBaremetalVolume</i>	<i>method</i> ), 115
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.CreateBaremetalPort</i>	<i>method</i> ), 110	<i>client.osc.v1.baremetal_volume_connector.UnsetBaremetalVolume</i>	<i>method</i> ), 115
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.DeleteBaremetalPort</i>	<i>method</i> ), 111	<i>client.osc.v1.baremetal_volume_target.CreateBaremetalVolume</i>	<i>method</i> ), 115
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.ListBaremetalPort</i>	<i>method</i> ), 111	<i>client.osc.v1.baremetal_volume_target.DeleteBaremetalVolume</i>	<i>method</i> ), 116
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.SetBaremetalPort</i>	<i>method</i> ), 111	<i>client.osc.v1.baremetal_volume_target.ListBaremetalVolumes</i>	<i>method</i> ), 116
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.ShowBaremetalPort</i>	<i>method</i> ), 111	<i>client.osc.v1.baremetal_volume_target.SetBaremetalVolume</i>	<i>method</i> ), 116
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_port.UnsetBaremetalPort</i>	<i>method</i> ), 112	<i>client.osc.v1.baremetal_volume_target.ShowBaremetalVolume</i>	<i>method</i> ), 116
take_action()	( <i>ironic-</i>	take_action()	( <i>ironic-</i>
<i>client.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup</i>	<i>method</i> ), 112	<i>client.osc.v1.baremetal_volume_target.UnsetBaremetalVolume</i>	<i>method</i> ), 117
take_action()	( <i>ironic-</i>	tempdir()	( <i>in module ironic-</i>
<i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup</i>	<i>method</i> ), 112	( <i>in module ironic-</i>	<i>common.utils</i> ), 93
		to_dict()	( <i>ironic-</i>
take_action()	( <i>ironic-</i>	client.common.apiclient.base.Resource	<i>method</i> ), 76
<i>client.osc.v1.baremetal_portgroup.ListBaremetalPortGroups</i>	<i>method</i> ), 113	to_dict()	( <i>ironicclient.common.base.Resource</i>

method), 89

trait

- openstack-baremetal-node-add-trait
  - command line option, 23
- openstack-baremetal-node-remove-trait
  - command line option, 38

## U

Unauthorized, 86

UndeployBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 109

UnprocessableEntity, 87

UnrescueBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 109

UnsetBaremetalAllocation (class in *ironicclient.osc.v1.baremetal\_allocation*), 94

UnsetBaremetalChassis (class in *ironic-client.osc.v1.baremetal\_chassis*), 96

UnsetBaremetalDeployTemplate (class in *ironic-client.osc.v1.baremetal\_deploy\_template*), 98

UnsetBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 109

UnsetBaremetalPort (class in *ironic-client.osc.v1.baremetal\_port*), 111

UnsetBaremetalPortGroup (class in *ironic-client.osc.v1.baremetal\_portgroup*), 113

UnsetBaremetalVolumeConnector (class in *ironic-client.osc.v1.baremetal\_volume\_connector*), 115

UnsetBaremetalVolumeTarget (class in *ironic-client.osc.v1.baremetal\_volume\_target*), 116

UnsupportedMediaType, 87

UnsupportedVersion, 88

update () (*ironic-client.common.apiclient.base.CrudManager method*), 75

update () (*ironic-client.v1.allocation.AllocationManager method*), 119

update () (*ironic-client.v1.chassis.ChassisManager method*), 121

update () (*ironic-client.v1.deploy\_template.DeployTemplateManager method*), 127

update () (*ironic-client.v1.driver.DriverManager method*), 128

update () (*ironicclient.v1.node.NodeManager method*), 136

update () (*ironicclient.v1.port.PortManager method*), 139

update () (*ironic-client.v1.portgroup.PortgroupManager method*), 141

update () (*ironic-client.v1.volume\_connector.VolumeConnectorManager method*), 143

update () (*ironic-client.v1.volume\_target.VolumeTargetManager method*), 144

## V

validate () (*ironic-client.v1.node.NodeManager method*), 136

ValidateBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 109

ValidationError, 88

vendor\_passthru () (*ironic-client.v1.driver.DriverManager method*), 128

vendor\_passthru () (*ironic-client.v1.node.NodeManager method*), 136

VersionNegotiationMixin (class in *ironic-client.common.http*), 89

vif\_attach () (*ironic-client.v1.node.NodeManager method*), 136

vif\_detach () (*ironic-client.v1.node.NodeManager method*), 136

vif\_list () (*ironic-client.v1.node.NodeManager method*), 137

vif-id

- openstack-baremetal-node-vif-attach
  - command line option, 48
- openstack-baremetal-node-vif-detach
  - command line option, 48

VifAttachBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 109

VifDetachBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 110

VifListBaremetalNode (class in *ironic-client.osc.v1.baremetal\_node*), 110

volume connector  
  openstack-baremetal-volume-connector-delete  
    command line option, [61](#)  
  openstack-baremetal-volume-connector-set  
    command line option, [62](#)  
  openstack-baremetal-volume-connector-unset  
    command line option, [64](#)  
volume target  
  openstack-baremetal-volume-target-delete  
    command line option, [65](#)  
  openstack-baremetal-volume-target-set  
    command line option, [67](#)  
  openstack-baremetal-volume-target-unset  
    command line option, [68](#)  
VolumeConnector (class in *ironic-client.v1.volume\_connector*), [142](#)  
VolumeConnectorManager (class in *ironic-client.v1.volume\_connector*), [142](#)  
VolumeTarget (class in *ironic-client.v1.volume\_target*), [143](#)  
VolumeTargetManager (class in *ironic-client.v1.volume\_target*), [143](#)

## W

wait() (ironic-client.v1.allocation.AllocationManager method), [119](#)  
wait\_for\_provision\_state() (ironic-client.v1.node.NodeManager method), [137](#)  
with\_retries() (in module *ironic-client.common.http*), [90](#)