
Python IroniC Client Documentation

Release 6.2.1.dev1

OpenStack IroniC Project

Jul 06, 2026

CONTENTS

1	Contents	3
1.1	ironicclient Python API	3
1.2	python-ironicclient User Documentation	4
1.3	Creating the Bare Metal service resources from file	88
1.4	python-ironicclient Contributor Documentation	91
1.5	Full Ironic Client Python API Reference	92
2	Indices and tables	211
	Python Module Index	213
	Index	215

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.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 `ironicclient.v1.client.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.common.apiclient.exceptions.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
```

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]
```

(continues on next page)

(continued from previous page)

```
[--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]
  [--sort-ascending | --sort-descending]
  [--limit <limit>]
  [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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]
  [--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]
  [--sort-ascending | --sort-descending]
  [--fields <field> [<field> ...]]
```

(continues on next page)

(continued from previous page)

```
[--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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 conductor

baremetal conductor list

List baremetal conductors

```
openstack baremetal conductor 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]
```

(continues on next page)

(continued from previous page)

```
[--sort-ascending | --sort-descending]
[--limit <limit>]
[--marker <conductor>]
[--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--limit <limit>

Maximum number of conductors to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.

--marker <conductor>

Hostname of the conductor (for example, of the last conductor in the list from a previous request). Returns the list of conductors after this conductor.

--sort <key>[:<direction>]

Sort output by specified conductor fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.

--long

Show detailed information about the conductors.

--fields <field>

One or more conductor 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 conductor show

Show baremetal conductor details

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

-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 conductor fields. Only these fields will be fetched from the server.

conductor

Hostname of the conductor

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 deploy template

baremetal deploy template create

Create a new deploy template

```
openstack baremetal deploy template create
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
  [--max-width <integer>]
  [--fit-width]
  [--print-empty]
  [--uuid <uuid>]
  [--extra <key=value>]
  --steps <steps>
  <name>
```

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

the output format, defaults to table

-c COLUMN, **--columnn 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.

--uuid <uuid>

UUID of the deploy template.

--extra <key=value>

Record arbitrary key/value metadata. Can be specified multiple times.

--steps <steps>

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

name

Unique name for this deploy template. Must be a valid trait name

This command is provided by the python-ironicclient plugin.

baremetal deploy template delete

Delete deploy template(s).

```
openstack baremetal deploy template delete <template> [<template> ...]
```

template

Name(s) or UUID(s) of the deploy template(s) to delete.

This command is provided by the python-ironicclient plugin.

baremetal deploy template list

List baremetal deploy templates.

```
openstack baremetal deploy template 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]
  [--sort-ascending | --sort-descending]
  [--limit <limit>]
  [--marker <template>]
  [--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
- sort-ascending**
sort the column(s) in ascending order
- sort-descending**
sort the column(s) in descending order
- limit** <limit>
Maximum number of deploy templates to return per request, 0 for no limit. Default is the maximum number used by the Baremetal API Service.
- marker** <template>
DeployTemplate UUID (for example, of the last deploy template in the list from a previous request). Returns the list of deploy templates after this UUID.
- sort** <key>[:<direction>]
Sort output by specified deploy template fields and directions (asc or desc) (default: asc). Multiple fields and directions can be specified, separated by comma.
- long**
Show detailed information about deploy templates.
- fields** <field>
One or more deploy template 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 deploy template set

Set baremetal deploy template properties.

```
openstack baremetal deploy template set
  [--name <name>]
  [--steps <steps>]
```

(continues on next page)

(continued from previous page)

```
[--extra <key=value>]
<template>
```

--name <name>

Set unique name of the deploy template. Must be a valid trait name.

--steps <steps>

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

--extra <key=value>

Extra to set on this baremetal deploy template (repeat option to set multiple extras).

template

Name or UUID of the deploy template

This command is provided by the python-ironicclient plugin.

baremetal deploy template show

Show baremetal deploy template details.

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

-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 deploy template fields. Only these fields will be fetched from the server.

template

Name or UUID of the deploy template.

This command is provided by the python-ironicclient plugin.

baremetal deploy template unset

Unset baremetal deploy template properties.

```
openstack baremetal deploy template unset [--extra <key>] <template>
```

--extra <key>

Extra to unset on this baremetal deploy template (repeat option to unset multiple extras).

template

Name or UUID of the deploy template

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]
  [--sort-ascending | --sort-descending]
  [--type <type>]
  [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--type <type>

Type of driver (classic or dynamic). The default is to list all of them.

--long

Show detailed information about the drivers.

--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 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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending
sort the column(s) in ascending order

--sort-descending
sort the column(s) in descending order

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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

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]
  [--fields <field> [<field> ...]]
  <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.

--fields <field>

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

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> ...]
```

node

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

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]
  [--sort-ascending | --sort-descending]
  [--long | --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

--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--long

Show detailed information about the BIOS settings.

--fields <field>

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

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>
  [<node> ...]
  <device>
```

--persistent

Make changes persistent for all future boots

node

Names or UUIDs of the nodes

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 boot mode set

Set the boot mode for the next baremetal node deployment

```
openstack baremetal node boot mode set <node> [<node> ...] <boot_mode>
```

node

Names or UUIDs of the nodes.

boot_mode

The boot mode to set for node (uefi/bios)

This command is provided by the python-ironicclient plugin.

baremetal node children list

Get a list of nodes associated as children.

```
openstack baremetal node children list
  [-f {json,shell,table,value,yaml}]
  [-c COLUMN]
  [--noindent]
  [--prefix PREFIX]
```

(continues on next page)

(continued from previous page)

```
[--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 clean

Set provision state of baremetal node to clean

```
openstack baremetal node clean
  [--wait [<time-out>]]
  (--clean-steps <clean-steps> | --runbook <runbook>)
  [--disable-ramdisk]
  <node>
  [<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. May be the path to a YAML file containing the clean steps; OR -, with the clean steps being read from standard input; OR a JSON string. The value should be a list of clean-step dictionaries; each dictionary should have keys interface and step, and optional key args.

--runbook <runbook>

The identifier of a predefined runbook to use for cleaning.

--disable-ramdisk

ironic-python-agent will not be booted for cleaning. Only steps explicitly marked as not requiring ironic-python-agent can be executed with this set.

node

Names or UUIDs of the nodes.

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> ...]
```

node

Names or UUIDs of the nodes

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> ...]
```

node

Names or UUIDs of the nodes

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>]
  [--instance-name <instance_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>]
```

(continues on next page)

(continued from previous page)

```
[--storage-interface <storage_interface>]
[--vendor-interface <vendor_interface>]
[--resource-class <resource_class>]
[--conductor-group <conductor_group>]
[--automated-clean | --no-automated-clean]
[--owner <owner>]
[--lessee <lessee>]
[--description <description>]
[--shard <shard>]
[--parent-node <parent_node>]
[--firmware-interface <firmware_interface>]
[--disable-power-off]
```

-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.

--instance-name <instance_name>

Name of the instance deployed on this 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 YAML 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

--no-automated-clean
Explicitly disable automated cleaning for the node

--owner <owner>
Owner of the node.

--lessee <lessee>
Lessee of the node.

--description <description>
Description for the node.

--shard <shard>
Shard for the node.

--parent-node <parent_node>
Parent node for the node being created.

--firmware-interface <firmware_interface>
Firmware interface used by the nodes driver. This is only applicable when the specified driver is a hardware type.

--disable-power-off
Explicitly disable power off actions on 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>]
  [--deploy-steps <deploy-steps>]
  <node>
  [<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 OR the path to the JSON file. 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* or a JSON file, a config drive will be generated on the server side (see the bare metal API reference for more details).

--deploy-steps <deploy-steps>

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

node

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node firmware list

List all Firmware Components of a node

```
openstack baremetal node firmware 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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

baremetal node history get

Get history event for a baremetal node.

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

-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.

event

UUID of the event.

This command is provided by the python-ironicclient plugin.

baremetal node history list

Get history events for a baremetal node.

```
openstack baremetal node history 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]
  [--sort-ascending | --sort-descending]
  [--long]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--long

Show detailed information about the node history events.

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> ...]
```

node

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node inventory save

Get hardware inventory of a node (in JSON format) or save it to file.

```
openstack baremetal node inventory save [--file <filename>] <node>
```

--file <filename>

Save inspection data to file with name (default: stdout).

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]
  [--sort-ascending | --sort-descending]
  [--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>]
  [--instance-name <instance_name>]
  [--sharded | --unsharded]
  [--shards <shards> [<shards> ...]]
```

(continues on next page)

(continued from previous page)

```
[--long | --fields <field> [<field> ...]]  
[--include-children | --parent-node <parent_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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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>

--instance-name <instance_name>

Filter the list of returned nodes by an instance name.

--sharded

List only nodes that are sharded.

--unsharded

List only nodes that are not sharded.

--shards <shards>

List only nodes that are in shards <shards>.

--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.

--include-children

Include children in the node list.

--parent-node <parent_node>

List only nodes associated with a parent node.

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>
  [<node> ...]
```

--reason <reason>

Reason for setting maintenance mode.

node

Names or UUIDs of the nodes.

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> ...]
```

node

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

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>
  [<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

Names or UUIDs of the nodes.

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>
  [<node> ...]
```

--power-timeout <power-timeout>

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

node

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

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>]
  [--deploy-steps <deploy-steps>]
  <node>
  [<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 OR the path to the JSON file. 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* or a JSON file, a config drive will be generated on the server side (see the bare metal API reference for more details).

--deploy-steps <deploy-steps>

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

node

Names or UUIDs of the nodes.

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> ...]
```

--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>
  [<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

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node secure boot off

Turn secure boot off

```
openstack baremetal node secure boot off <node> [<node> ...]
```

node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

baremetal node secure boot on

Turn secure boot on

```
openstack baremetal node secure boot on <node> [<node> ...]
```

node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

baremetal node service

Set provision state of baremetal node to service

```
openstack baremetal node service
  [--wait [<time-out>]]
  (--service-steps <service-steps> | --runbook <runbook>)
  [--disable-ramdisk]
```

(continues on next page)

(continued from previous page)

```
<node>
[<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.

--service-steps <service-steps>

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

--runbook <runbook>

The identifier of a predefined runbook to use for servicing.

--disable-ramdisk

ironic-python-agent will not be booted for cleaning. Only steps explicitly marked as not requiring ironic-python-agent can be executed with this set.

node

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node set

Set baremetal properties

```
openstack baremetal node set
  [--instance-uuid <uuid>]
  [--instance-name <instance_name>]
  [--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]
  [--firmware-interface <firmware_interface> | --reset-firmware-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]
```

(continues on next page)

(continued from previous page)

```
[--resource-class <resource_class>]
[--conductor-group <conductor_group>]
[--automated-clean | --no-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>]
[--shard <shard>]
[--parent-node <parent_node>]
[--enable-power-off | --disable-power-off]
<node>
[<node> ...]
```

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

--instance-name <instance_name>
Set the name of the instance deployed on this node.

--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

--firmware-interface <firmware_interface>

Set the firmware interface for the node

--reset-firmware-interface

Reset the firmware interface for 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 YAML 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.

--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

--no-automated-clean

Explicitly disable 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 YAML 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
- shard** <shard>
Set the shard for the node
- parent-node** <parent_node>
Set the parent node for the node
- enable-power-off**
Explicitly enable power off actions on nodes
- disable-power-off**
Explicitly disable power off actions on nodes

node

Names or UUIDs of the nodes.

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 add

Add traits to a node.

```
openstack baremetal node trait add <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 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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

node

Name or UUID of the node

This command is provided by the python-ironicclient plugin.

baremetal node trait remove

Remove trait(s) from a node.

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

Set provision state of baremetal node to deleted

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

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node unhold

Set provision state of baremetal node to unhold

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

node

Names or UUIDs of the nodes.

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>
  [<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

Names or UUIDs of the nodes.

This command is provided by the python-ironicclient plugin.

baremetal node unset

Unset baremetal properties

```
openstack baremetal node unset
  [--instance-uuid]
  [--instance-name]
  [--name]
  [--resource-class]
```

(continues on next page)

(continued from previous page)

```
[--target-raid-config]
[--property <key>]
[--extra <key>]
[--driver-info <key>]
[--instance-info <key>]
[--chassis-uuid]
[--bios-interface]
[--boot-interface]
[--console-interface]
[--deploy-interface]
[--firmware-interface]
[--inspect-interface]
[--network-data]
[--management-interface]
[--network-interface]
[--power-interface]
[--raid-interface]
[--rescue-interface]
[--storage-interface]
[--vendor-interface]
[--conductor-group]
[--automated-clean]
[--protected]
[--protected-reason]
[--retired]
[--retired-reason]
[--owner]
[--lessee]
[--description]
[--shard]
[--parent-node]
<node>
[<node> ...]
```

--instance-uuid

Unset instance UUID on this baremetal node

--instance-name

Unset instance name 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 items in driver information)

--instance-info <key>

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

--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

--firmware-interface

Unset firmware 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

--shard

Unset the shard field of the node

--parent-node

Unset the parent node field of the node

node

Names or UUIDs of the nodes.

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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

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>]
  [--portgroup-uuid <portgroup-uuid>]
  [--vif-info <key=value>]
  <node>
  <vif-id>
```

--port-uuid <port-uuid>

UUID of the baremetal port to attach the VIF to.

--portgroup-uuid <portgroup-uuid>

UUID of the baremetal portgroup 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]
  [--sort-ascending | --sort-descending]
  <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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

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>]
  [--pxe-enabled <boolean>]
  [--port-group <uuid>]
  [--physical-network <physical network>]
  [--is-smartnic]
  [--name <port name>]
  [--description <port description>]
  [--vendor <port vendor>]
  [--category <port category>]
  <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.
- 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
- name <port name>**
Name of the port.
- description <port description>**
An optional description for the port.

--vendor <port vendor>

An optional vendor for the port.

--category <port category>

An optional category for the 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]
  [--physical-network <physical network>]
  [--category <port category>]
```

-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.

--physical-network <physical network>

Name of the physical network to which the ports in this port group are connected.

--category <port category>

An optional category for the port group.

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]
  [--sort-ascending | --sort-descending]
  [--limit <limit>]
  [--marker <port group>]
  [--sort <key>[:<direction>]]
  [--address <mac-address>]
  [--node <node>]
  [--shards <shard> [<shard> ...]]
  [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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).

--shards <shard>

Only list port groups of nodes in these shards.

--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]
  [--physical-network <physical network>]
  [--category <category>]
  <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.

--physical-network <physical network>

Set a physical network for the ports in this port group

--category <category>

Set a category for this port group

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>]
  [--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>]
  [--physical-network]
  [--category]
  <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).

--physical-network

Unset the physical network on this baremetal port group.

--category

Unset the category for this port group.

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]
  [--sort-ascending | --sort-descending]
  [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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>]
```

(continues on next page)

(continued from previous page)

```
[--port-group <uuid>]
[--local-link-connection <key=value>]
[--pxe-enabled | --pxe-disabled]
[--physical-network <physical network>]
[--is-smartnic]
[--name <name>]
[--description <description>]
[--vendor <vendor>]
[--category <category>]
<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

--name <name>

Set name for this port

--description <description>

Set a description for this port

--vendor <vendor>

Set a vendor for this port

--category <category>

Set a category for this 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>]
  [--local-link-connection <key>]
  [--port-group]
  [--physical-network]
  [--is-smartnic]
  [--name]
  [--description]
  [--vendor]
  [--category]
  <port>
```

--extra <key>

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

--local-link-connection <key>

Remove local link connection on this port (repeat option to unset multiple keys)

--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

--name

Unset the name for this port

--description

Unset the description for this port.

--vendor

Unset the vendor for this port.

--category

Unset the category for this 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]
  [--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]
    [--sort-ascending | --sort-descending]
    [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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]
  [--sort-ascending | --sort-descending]
  [--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

--sort-ascending

sort the column(s) in ascending order

--sort-descending

sort the column(s) in descending order

--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
  [--node <uuid>]
  [--type <volume type>]
  [--property <key=value>]
  [--boot-index <boot index>]
  [--volume-id <volume id>]
  [--extra <key=value>]
  <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: Client, filenames:
                                                  list[str]) → None
```

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": "ipmi",
          "portgroups": [
            {
              "name": "switch.cz7882.ports.1-2",
              "ports": [

```

(continues on next page)

(continued from previous page)

```
        "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"
    }
],
"driver_info": {
    "ipmi_address": "192.168.1.23",
    "ipmi_username": "BmcUsername",
    "ipmi_password": "BmcPassword",
}
},
{
    "name": "node-4",
    "driver": "ipmi",
    "ports": [
        {
            "address": "00:00:00:00:00:04"
        },
        {
            "address": "00:00:00:00:00:01"
        }
    ]
}
]
},
"nodes": [
    {
        "name": "node-5",
        "driver": "ipmi",
        "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": "ipmi"
    }
]
```

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

#openstack-ironic on OFTC IRC Network

There is a very active chat channel at <irc://irc.oftc.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/>.

Developer Certificate of Origin

In order to contribute to the python-ironicclient project, you need to adhere to the [Developer Certificate of Origin](#). OpenStack utilizes the Developer Certificate of Origin (DCO) as a lightweight means to confirm that you are entitled to contribute the code you submit. This ensures that you are providing your contributions under the projects license and that you have the right to do so.

See also

- <https://docs.openstack.org/contributors/common/dco.html>

Project Hosting Details

Bug tracker

<https://bugs.launchpad.net/python-ironicclient>

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 tests have been removed as of November 2024.

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.ManagerProtocol(*args, **kwargs)
```

Bases: Protocol

Minimal protocol for Resources manager dependency.

client: Any

```
class ironicclient.common.apiclient.base.Resource(manager: ManagerProtocol, info: dict[str, Any], loaded: bool = False)
```

Bases: object

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

This is pretty much just a bag for attributes.

HUMAN_ID: bool = False

NAME_ATTR: str = 'name'

get() → None

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: str | None

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

is_loaded() → bool

set_loaded(val: bool) → None

to_dict() → dict[str, Any]

ironicclient.common.apiclient.exceptions module

Exception definitions.

```
exception ironicclient.common.apiclient.exceptions.AmbiguousEndpoints(endpoints: object | None = None)
```

Bases: [EndpointException](#)

Found more than one matching endpoint in Service Catalog.

exception `ironicclient.common.apiclient.exceptions.AuthPluginOptionsMissing`(*opt_names: list[str]*)

Bases: [`AuthorizationFailure`](#)

Auth plugin misses some options.

exception `ironicclient.common.apiclient.exceptions.AuthSystemNotFound`(*auth_system: str*)

Bases: [`AuthorizationFailure`](#)

User has specified an AuthSystem that is not installed.

exception `ironicclient.common.apiclient.exceptions.AuthorizationFailure`

Bases: [`ClientException`](#)

Cannot authorize API client.

exception `ironicclient.common.apiclient.exceptions.BadGateway`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [`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: int = 502

message: str = 'Bad Gateway'

exception `ironicclient.common.apiclient.exceptions.BadRequest`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [`HTTPClientError`](#)

HTTP 400 - Bad Request.

The request cannot be fulfilled due to bad syntax.

```
http_status:  int = 400
```

```
message:  str = '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: `ClientException`

Error in CLI tool.

exception `ironicclient.common.apiclient.exceptions.Conflict`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: `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:  int = 409
```

```
message:  str = 'Conflict'
```

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

Bases: `ClientException`

Cannot connect to API service.

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

Bases: `ConnectionError`

Connection refused while trying to connect to API service.

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

Bases: `ClientException`

Something is rotten in Service Catalog.

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

Bases: `EndpointException`

Could not find requested endpoint in Service Catalog.

```
exception ironicclient.common.apiclient.exceptions.ExpectationFailed(message: str
                                                                    | None =
                                                                    None,
                                                                    details: str |
                                                                    None =
                                                                    None,
                                                                    response:
                                                                    Response |
                                                                    None =
                                                                    None,
                                                                    request_id:
                                                                    str | None =
                                                                    None, url:
                                                                    str | None =
                                                                    None,
                                                                    method: str |
                                                                    None =
                                                                    None,
                                                                    http_status:
                                                                    int | None =
                                                                    None)
```

Bases: [*HTTPClientError*](#)

HTTP 417 - Expectation Failed.

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

http_status: int = 417

message: str = 'Expectation Failed'

```
exception ironicclient.common.apiclient.exceptions.Forbidden(message: str | None =
                                                                    None, details: str |
                                                                    None = None,
                                                                    response: Response |
                                                                    None = None,
                                                                    request_id: str | None
                                                                    = None, url: str | None
                                                                    = None, method: str |
                                                                    None = None,
                                                                    http_status: int | None
                                                                    = None)
```

Bases: [*HTTPClientError*](#)

HTTP 403 - Forbidden.

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

http_status: int = 403

message: str = 'Forbidden'

```
exception ironicclient.common.apiclient.exceptions.GatewayTimeout(message: str |  
                                                                    None = None,  
                                                                    details: str |  
                                                                    None = None,  
                                                                    response:  
                                                                    Response | None  
                                                                    = None,  
                                                                    request_id: str |  
                                                                    None = None,  
                                                                    url: str | None =  
                                                                    None, method:  
                                                                    str | None =  
                                                                    None,  
                                                                    http_status: int |  
                                                                    None = None)
```

Bases: [*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: int = 504

message: str = 'Gateway Timeout'

```
exception ironicclient.common.apiclient.exceptions.Gone(message: str | None = None,  
                                                           details: str | None = None,  
                                                           response: Response | None =  
                                                           None, request_id: str | None  
                                                           = None, url: str | None =  
                                                           None, method: str | None =  
                                                           None, http_status: int | None  
                                                           = None)
```

Bases: [*HTTPClientError*](#)

HTTP 410 - Gone.

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

http_status: int = 410

message: str = 'Gone'

```
exception ironicclient.common.apiclient.exceptions.HTTPClientError(message: str |  
                                                                    None = None,  
                                                                    details: str |  
                                                                    None = None,  
                                                                    response:  
                                                                    Response |  
                                                                    None = None,  
                                                                    request_id: str  
                                                                    | None = None,  
                                                                    url: str | None  
                                                                    = None,  
                                                                    method: str |  
                                                                    None = None,  
                                                                    http_status: int  
                                                                    | None =  
                                                                    None)
```

Bases: [HttpError](#)

Client-side HTTP error.

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

message: **str** = 'HTTP Client Error'

```
exception ironicclient.common.apiclient.exceptions.HTTPRedirection(message: str |  
                                                                    None = None,  
                                                                    details: str |  
                                                                    None = None,  
                                                                    response:  
                                                                    Response |  
                                                                    None = None,  
                                                                    request_id: str  
                                                                    | None = None,  
                                                                    url: str | None  
                                                                    = None,  
                                                                    method: str |  
                                                                    None = None,  
                                                                    http_status: int  
                                                                    | None =  
                                                                    None)
```

Bases: [HttpError](#)

HTTP Redirection.

message: **str** = 'HTTP Redirection'

```
exception ironicclient.common.apiclient.exceptions.HttpError(message: str | None =  
None, details: str |  
None = None,  
response: Response |  
None = None,  
request_id: str | None  
= None, url: str | None  
= None, method: str |  
None = None,  
http_status: int | None  
= None)
```

Bases: [*ClientException*](#)

The base exception class for all HTTP exceptions.

```
http_status:  int = 0
```

```
message:  str = 'HTTP Error'
```

```
exception ironicclient.common.apiclient.exceptions.HttpNotImplemented(message:  
str | None  
= None,  
details: str  
| None =  
None,  
response:  
Response |  
None =  
None,  
request_id:  
str | None  
= None,  
url: str |  
None =  
None,  
method:  
str | None  
= None,  
http_status:  
int | None  
= None)
```

Bases: [*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:  int = 501
```

```
message:  str = 'Not Implemented'
```

```
exception ironicclient.common.apiclient.exceptions.HttpServerError(message: str |  
                                                                    None = None,  
                                                                    details: str |  
                                                                    None = None,  
                                                                    response:  
                                                                    Response |  
                                                                    None = None,  
                                                                    request_id: str  
                                                                    | None = None,  
                                                                    url: str | None  
                                                                    = None,  
                                                                    method: str |  
                                                                    None = None,  
                                                                    http_status: int  
                                                                    | None =  
                                                                    None)
```

Bases: [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: str = 'HTTP Server Error'

```

exception ironicclient.common.apiclient.exceptions.HttpVersionNotSupported(message:
    str |
    None
    =
    None,
    de-
    tails:
    str |
    None
    =
    None,
    re-
    sponse:
    Re-
    sponse
    |
    None
    =
    None,
    re-
    quest_id:
    str |
    None
    =
    None,
    url:
    str |
    None
    =
    None,
    method:
    str |
    None
    =
    None,
    http_status:
    int |
    None
    =
    None)

```

Bases: [HttpServerError](#)

HTTP 505 - HttpVersion Not Supported.

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

http_status: `int = 505`

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

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

Bases: [*HttpServerError*](#)

HTTP 500 - Internal Server Error.

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

http_status: **int** = 500

message: **str** = 'Internal Server Error'

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

Bases: [*HTTPClientError*](#)

HTTP 411 - Length Required.

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

http_status: **int** = 411

message: `str` = 'Length Required'

exception `ironicclient.common.apiclient.exceptions.MethodNotAllowed`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*HTTPClientError*](#)

HTTP 405 - Method Not Allowed.

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

http_status: `int` = 405

message: `str` = 'Method Not Allowed'

exception `ironicclient.common.apiclient.exceptions.MultipleChoices`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*HTTPRedirection*](#)

HTTP 300 - Multiple Choices.

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

http_status: `int` = 300

message: `str` = 'Multiple Choices'

exception `ironicclient.common.apiclient.exceptions.NoUniqueMatch`

Bases: [*ClientException*](#)

Multiple entities found instead of one.

exception `ironicclient.common.apiclient.exceptions.NotAcceptable`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*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: `int = 406`

message: `str = 'Not Acceptable'`

exception `ironicclient.common.apiclient.exceptions.NotFound`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*HTTPClientError*](#)

HTTP 404 - Not Found.

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

http_status: `int = 404`

message: `str = 'Not Found'`

```
exception ironicclient.common.apiclient.exceptions.PaymentRequired(message: str |  
                                                                    None = None,  
                                                                    details: str |  
                                                                    None = None,  
                                                                    response:  
                                                                    Response |  
                                                                    None = None,  
                                                                    request_id: str  
                                                                    | None = None,  
                                                                    url: str | None  
                                                                    = None,  
                                                                    method: str |  
                                                                    None = None,  
                                                                    http_status: int  
                                                                    | None =  
                                                                    None)
```

Bases: [*HTTPClientError*](#)

HTTP 402 - Payment Required.

Reserved for future use.

http_status: int = 402

message: str = 'Payment Required'

```
exception ironicclient.common.apiclient.exceptions.PreconditionFailed(message:  
                                                                    str | None  
                                                                    = None,  
                                                                    details: str  
                                                                    | None =  
                                                                    None,  
                                                                    response:  
                                                                    Response |  
                                                                    None =  
                                                                    None,  
                                                                    request_id:  
                                                                    str | None  
                                                                    = None,  
                                                                    url: str |  
                                                                    None =  
                                                                    None,  
                                                                    method:  
                                                                    str | None  
                                                                    = None,  
                                                                    http_status:  
                                                                    int | None  
                                                                    = None)
```

Bases: [*HTTPClientError*](#)

HTTP 412 - Precondition Failed.

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

http_status: int = 412

message: str = 'Precondition Failed'

exception ironicclient.common.apiclient.exceptions.**ProxyAuthenticationRequired**(*message:*
str
|
None
=
None,
de-
tails:
str
|
None
=
None,
re-
sponse:
Re-
sponse
|
None
=
None,
re-
quest_id:
str
|
None
=
None,
url:
str
|
None
=
None,
method:
str
|
None
=
None,
http_status:
int
|
None
=
None)

Bases: *HTTPClientError*

HTTP 407 - Proxy Authentication Required.

The client must first authenticate itself with the proxy.

http_status: `int = 407`

message: `str = 'Proxy Authentication Required'`

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

Bases: [*HTTPClientError*](#)

HTTP 413 - Request Entity Too Large.

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

http_status: `int = 413`

message: `str = 'Request Entity Too Large'`

```
exception ironicclient.common.apiclient.exceptions.RequestTimeout(message: str |
    None = None,
    details: str |
    None = None,
    response:
    Response | None
    = None,
    request_id: str |
    None = None,
    url: str | None =
    None, method:
    str | None =
    None,
    http_status: int |
    None = None)
```

Bases: [*HTTPClientError*](#)

HTTP 408 - Request Timeout.

The server timed out waiting for the request.

http_status: `int = 408`

message: `str = 'Request Timeout'`

```
exception ironicclient.common.apiclient.exceptions.RequestUriTooLong(message: str
| None =
None,
details: str |
None =
None,
response:
Response |
None =
None,
request_id:
str | None =
None, url:
str | None =
None,
method: str |
None =
None,
http_status:
int | None =
None)
```

Bases: *HTTPClientError*

HTTP 414 - Request-URI Too Long.

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

http_status: int = 414

message: str = 'Request-URI Too Long'

```

exception ironicclient.common.apiclient.exceptions.RequestedRangeNotSatisfiable(message:
    str
    |
    None
    =
    None,
    de-
    tails:
    str
    |
    None
    =
    None,
    re-
    sponse:
    Re-
    sponse
    |
    None
    =
    None,
    re-
    quest_id:
    str
    |
    None
    =
    None,
    url:
    str
    |
    None
    =
    None,
    method:
    str
    |
    None
    =
    None,
    http_status:
    int
    |
    None
    =
    None)

```

Bases: [*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: `int = 416`

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

exception `ironicclient.common.apiclient.exceptions.ServiceUnavailable`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*HttpServerError*](#)

HTTP 503 - Service Unavailable.

The server is currently unavailable.

http_status: `int` = 503

message: `str` = 'Service Unavailable'

exception `ironicclient.common.apiclient.exceptions.Unauthorized`(*message: str | None = None, details: str | None = None, response: Response | None = None, request_id: str | None = None, url: str | None = None, method: str | None = None, http_status: int | None = None*)

Bases: [*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: int = 401
```

```
message: str = 'Unauthorized'
```

```
exception ironicclient.common.apiclient.exceptions.UnprocessableEntity(message:  
    str | None  
    = None,  
    details:  
    str | None  
    = None,  
    response:  
    Response  
    | None =  
    None, re-  
    quest_id:  
    str | None  
    = None,  
    url: str |  
    None =  
    None,  
    method:  
    str | None  
    = None,  
    http_status:  
    int | None  
    = None)
```

Bases: [*HTTPClientError*](#)

HTTP 422 - Unprocessable Entity.

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

```
http_status: int = 422
```

```
message: str = 'Unprocessable Entity'
```

```
exception ironicclient.common.apiclient.exceptions.UnsupportedMediaType(message:
    str |
    None =
    None,
    details:
    str |
    None =
    None,
    re-
    sponse:
    Re-
    sponse |
    None =
    None,
    re-
    quest_id:
    str |
    None =
    None,
    url: str |
    None =
    None,
    method:
    str |
    None =
    None,
    http_status:
    int |
    None =
    None)
```

Bases: [*HTTPClientError*](#)

HTTP 415 - Unsupported Media Type.

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

http_status: **int** = 415

message: **str** = 'Unsupported Media Type'

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

Bases: [*ClientException*](#)

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

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

Bases: [*ClientException*](#)

Error in validation on API client side.

```
ironicclient.common.apiclient.exceptions.from_response(response: Response, method:
    str, url: str) → HttpError
```

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*: `SessionClient`)

Bases: `Manager`[`ResourceT`]

Provides creation operations with a particular API.

create(*os_ironic_api_version*: `str` | `None` = `None`, *global_request_id*: `str` | `None` = `None`,
***kwargs*: `Any`) → `ResourceT` | `None`

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*: `SessionClient`)

Bases: `Generic`[`ResourceT`]

Provides CRUD operations with a particular API.

abstract property resource_class: `type`[`ResourceT`]

The resource class

class `ironicclient.common.base.Resource`(*manager*: `ManagerProtocol`, *info*: `dict`[`str`, `Any`],
loaded: `bool` = `False`)

Bases: `Resource`

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

This is pretty much just a bag for attributes.

to_dict() → `dict`[`str`, `Any`]

`ironicclient.common.base.getid(obj: str | Resource) → str`

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: str, port: str | int, expiry: int | None = None) → str | 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: str, port: str | int, data: str) → None`

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: str | list[str],
api_version_select_state: str, max_retries:
int | None, retry_interval: int | None,
**kwargs: Any)`

Bases: `VersionNegotiationMixin`, `LegacyJsonAdapter`

HTTP client based on Keystone client session.

`json_request(method: str, url: str, **kwargs: Any) → tuple[Response, dict[str, Any] | list[Any] | bytes | None]`

`raw_request(method: str, url: str, **kwargs: dict[str, str]) → Response`

`class ironicclient.common.http.VersionNegotiationMixin`

Bases: `object`

api_version_select_state: str

negotiate_version(conn: Session, resp: Response | None) → str

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

os_ironic_api_version: str | list[str]

`ironicclient.common.http.get_server(url: str | None) → tuple[str | None, str | None]`

Extract and return the server & port.

`ironicclient.common.http.with_retries(func: Callable[[...], _RT]) → Callable[[...], _RT]`

Wrapper for `_http_request` adding support for retries.

ironicclient.common.i18n module

ironicclient.common.utils module

class `ironicclient.common.utils.HashColumn(value: _T)`

Bases: `FormattableColumn[Any]`

human_readable() → `str | None`

Return a basic human readable version of the data.

class `ironicclient.common.utils.ListCommandArgs(*args, **kwargs)`

Bases: `Protocol`

Protocol for parsed args passed to list commands (marker, limit, etc).

detail: `bool`

fields: `list[list[str]] | None`

limit: `int | None`

marker: `str | None`

sort_dir: `str | None`

sort_key: `str | None`

`ironicclient.common.utils.args_array_to_dict(kwargs: dict[str, Any], key_to_convert: str) → dict[str, Any]`

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: str, attributes: list[str]) → list[dict[str, Any]]`

`ironicclient.common.utils.bool_argument_value(arg_name: str, bool_str: str, strict: bool = True, default: bool = False) → bool`

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_api_version_support(current_version: str | list[str], required_version: str) → bool`

Check if the current API version supports a required version.

Parameters

- **current_version** Current API version being used (e.g. 1.112)
- **required_version** Minimum API version required (e.g. 1.112)

Returns

`True` if the current version meets the requirement, `False` otherwise. Also returns `True` if the version format is invalid (let the server decide).

`ironicclient.common.utils.check_empty_arg(arg: str, arg_descriptor: str) → None`

`ironicclient.common.utils.check_for_invalid_fields(fields: list[str] | None, valid_fields: list[str]) → None`

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: str | None = None, limit: int | None = None, sort_key: str | None = None, sort_dir: str | None = None, fields: list[str] | None = None, detail: bool = False, project: str | None = None, public: bool | None = None) → list[str]`

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: ListCommandArgs, fields:
                                                list[str], field_labels: list[str]) →
                                                dict[str, Any]
```

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: dict[str,
                                                                    Any], props:
                                                                    list[str] | None =
                                                                    None) → dict[str,
                                                                    Any]
```

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.format_hash(data: dict[str, Any] | None, prefix: str = ") → str |
None
```

```
ironicclient.common.utils.get_from_stdin(info_desc: str) → str
```

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.get_json_data(data: bytes) → dict[str, Any] | list[Any] | None`

Check if the binary data is JSON and parse it if so.

Parameters

data Raw bytes to parse (e.g. response body).

Returns

Parsed dict or list if valid JSON, otherwise None.

`ironicclient.common.utils.handle_json_arg(json_arg: str | None, info_desc: str) → list[Any] | dict[str, Any] | None`

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, or None if json_arg is empty (e.g. after reading from stdin with no input).

Raises

InvalidAttribute if the argument cannot be parsed.

`ironicclient.common.utils.handle_json_or_file_arg(json_arg: str) → list[Any] | dict[str, Any]`

Attempts to read JSON argument from file or string.

Parameters

json_arg May be a file name containing the YAML or 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: list[str] | None) → dict[str, Any]`

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: str) → bytes`

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: int | float, poll_interval: int | float, poll_delay_function: Callable[[int | float], Any] | None, timeout_message: str | Callable[[], str]) → Iterator[int]`

`ironicclient.common.utils.split_and_deserialize(string: str) → tuple[str, Any]`

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(suffix: str | None = None, prefix: str | None = None, dir: str | None = None) → Generator[str, None, None]`

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,
    app_args:
    Namespace |
    None,
    cmd_name:
    str |
    None =
    None)
```

Bases: [ShowOne](#)

Create a new baremetal allocation.

get_parser(prog_name: str) → ArgumentParser
Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_allocation.CreateBaremetalAllocation
(WARNING)>
```

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocation(app:
    _app.App,
    app_args:
    Namespace |
    None,
    cmd_name:
    str |
    None =
    None)
```

Bases: *Command*

Unregister baremetal allocation(s).

get_parser(prog_name: str) → ArgumentParser
 Return an argparse.ArgumentParser.

log: logging.Logger = <Logger
 ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocation
 (WARNING)>

take_action(parsed_args: Namespace) → None
 Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocation(app: App,
    app_args:
    Namespace |
    None,
    cmd_name:
    str | None
    = None)
```

Bases: *Lister*

List baremetal allocations.

get_parser(prog_name: str) → ArgumentParser
 Return an argparse.ArgumentParser.

log: logging.Logger = <Logger
 ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocation
 (WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
 Run command.

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.App,
    app_args:
    Namespace
    | None,
    cmd_name:
    str | None
    = None)
```

Bases: [Command](#)

Set baremetal allocation properties.

get_parser(*prog_name: str*) → `ArgumentParser`
 Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`
ironicclient.osc.v1.baremetal_allocation.SetBaremetalAllocation (WARNING)>

take_action(*parsed_args: Namespace*) → `None`
 Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocation(app: App,
    app_args:
    Names-
    pace |
    None,
    cmd_name:
    str | None
    = None)
```

Bases: [ShowOne](#)

Show baremetal allocation details.

get_parser(*prog_name: str*) → `ArgumentParser`
 Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`
ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocation
 (WARNING)>

take_action(*parsed_args: Namespace*) → `tuple[tuple[str, ...], tuple[Any, ...]]`
 Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocation(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str |
                                                                    None =
                                                                    None)
```

Bases: [Command](#)

Unset baremetal allocation properties.

get_parser(prog_name: str) → ArgumentParser
Return an argparse.ArgumentParser.

log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocation
(WARNING)>

take_action(parsed_args: Namespace) → None
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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: [ShowOne](#)

Create a new chassis.

get_parser(prog_name: str) → ArgumentParser
Return an argparse.ArgumentParser.

log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_chassis.CreateBaremetalChassis (WARNING)>

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
Run command.
Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassis(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: *Command*

Delete a chassis.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassis (WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassis(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None, cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: *Lister*

List the chassis.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassis (WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App,
                                                                    app_args:
                                                                    Namespace | None,
                                                                    cmd_name: str |
                                                                    None = None)
```

Bases: *Command*

Set chassis properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_chassis.SetBaremetalChassis (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassis(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None, cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [ShowOne](#)

Show chassis details.

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassis (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
```

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassis(app: _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str |
                                                                    None = None)
```

Bases: [Command](#)

Unset chassis properties.

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassis (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Lister](#)

List baremetal conductors

```
get_parser(prog_name: str) → ArgumentParser
```

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
```

```
ironicclient.osc.v1.baremetal_conductor.ListBaremetalConductor (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
```

Run command.

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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [ShowOne](#)

Show baremetal conductor details

```
get_parser(prog_name: str) → ArgumentParser
```

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
```

```
ironicclient.osc.v1.baremetal_conductor.ShowBaremetalConductor (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
```

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

ironicclient.osc.v1.baremetal_create module

```
class ironicclient.osc.v1.baremetal_create.CreateBaremetal(app: _app.App,
                                                            app_args: Namespace |
                                                            None, cmd_name: str |
                                                            None = None)
```

Bases: [Command](#)

Create resources from files

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_create.CreateBaremetal` (WARNING)>

take_action(*parsed_args: Namespace*) → *None*

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.CreateBaremetalDeployTemplate(app:  
                                     App,  
                                     app_args:  
                                     Names-  
                                     pace  
                                     |  
                                     None,  
                                     cmd_name:  
                                     str  
                                     |  
                                     None  
                                     =  
                                     None)
```

Bases: [ShowOne](#)

Create a new deploy template

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger ironicclient.osc.v1.`

`baremetal_deploy_template.CreateBaremetalDeployTemplate` (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate(app:  
                                     _app.App,  
                                     app_args:  
                                     Names-  
                                     pace  
                                     |  
                                     None,  
                                     cmd_name:  
                                     str  
                                     |  
                                     None  
                                     =  
                                     None)
```

Bases: *Command*

Delete deploy template(s).

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger = <Logger ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplate(app:  
                                                                           App,  
                                                                           app_args:  
                                                                           Names-  
                                                                           pace  
                                                                           |  
                                                                           None,  
                                                                           cmd_name:  
                                                                           str  
                                                                           |  
                                                                           None  
                                                                           =  
                                                                           None)
```

Bases: *Lister*

List baremetal deploy templates.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger = <Logger ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplate (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [Command](#)

Set baremetal deploy template properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate
(WARNING)>
```

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate(app:
    App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [ShowOne](#)

Show baremetal deploy template details.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate
(WARNING)>
```

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Names-
                                                                    pace
                                                                    |
                                                                    None,
                                                                    cmd_name:
                                                                    str
                                                                    |
                                                                    None
                                                                    =
                                                                    None)
```

Bases: [Command](#)

Unset baremetal deploy template properties.

get_parser(*prog_name: str*) → ArgumentParser

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate`
(WARNING)>

take_action(*parsed_args: Namespace*) → None

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,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Lister](#)

List the enabled drivers.

get_parser(*prog_name: str*) → ArgumentParser

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_driver.ListBaremetalDriver` (WARNING)>

take_action(*parsed_args: Namespace*) → tuple[Sequence[str], Iterable[Any]]

Run command.

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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None
                                                                    = None)
```

Bases: [Lister](#)

List the driver properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverProperty
(WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str |
                                                                    None
                                                                    =
                                                                    None)
```

Bases: [Lister](#)

List a drivers RAID logical disk properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverRaidProperty
(WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None
                                                                    = None)
```

Bases: [ShowOne](#)

Call a vendor passthru method for a driver.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_driver.PassthruCallBaremetalDriver

(WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriver(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None
                                                                    = None)
```

Bases: [Lister](#)

List available vendor passthru methods for a driver.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriver

(WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [ShowOne](#)

Show information about a driver.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_driver.ShowBaremetalDriver` (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

ironicclient.osc.v1.baremetal_inspection_rule module

```
class ironicclient.osc.v1.baremetal_inspection_rule.CreateBaremetalInspectionRule(app:  
                                     App,  
                                     app_args:  
                                     Names-  
                                     pace  
                                     |  
                                     None,  
                                     cmd_name:  
                                     str  
                                     |  
                                     None  
                                     =  
                                     None)
```

Bases: [ShowOne](#)

Create a new rule

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger ironicclient.osc.v1.`

`baremetal_inspection_rule.CreateBaremetalInspectionRule` (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_inspection_rule.DeleteBaremetalInspectionRule(app:  
                                     _app.App,  
                                     app_args:  
                                     Names-  
                                     pace  
                                     |  
                                     None,  
                                     cmd_name:  
                                     str  
                                     |  
                                     None  
                                     =  
                                     None)
```

Bases: *Command*

Delete rule(s).

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger = <Logger ironicclient.osc.v1.baremetal_inspection_rule.DeleteBaremetalInspectionRule (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_inspection_rule.ListBaremetalInspectionRule(app:
                                                                    App,
                                                                    app_args:
                                                                    Names-
                                                                    pace
                                                                    |
                                                                    None,
                                                                    cmd_name:
                                                                    str
                                                                    |
                                                                    None
                                                                    =
                                                                    None)
```

Bases: *Lister*

List baremetal rules.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger = <Logger ironicclient.osc.v1.baremetal_inspection_rule.ListBaremetalInspectionRule (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_inspection_rule.SetBaremetalInspectionRule(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [Command](#)

Set baremetal rule properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_inspection_rule.SetBaremetalInspectionRule
(WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_inspection_rule.ShowBaremetalInspectionRule(app:
    App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [ShowOne](#)

Show baremetal rule details.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_inspection_rule.ShowBaremetalInspectionRule
(WARNING)>

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_inspection_rule.UnsetBaremetalInspectionRule(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [Command](#)

Unset baremetal rule properties.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_inspection_rule.UnsetBaremetalInspectionRule
(WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

ironicclient.osc.v1.baremetal_node module

```
class ironicclient.osc.v1.baremetal_node.AbortBaremetalNode(app: _app.App,
    app_args: Namespace |
    None, cmd_name: str |
    None = None)
```

Bases: [ProvisionStateBaremetalNode](#)

Set provision state of baremetal node to abort

PROVISION_STATE: str = 'abort'

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.AbortBaremetalNode (WARNING)>

```
class ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNode(app: _app.App,
    app_args:
    Namespace | None,
    cmd_name: str |
    None = None)
```

Bases: [Command](#)

Add traits to a node.

get_parser(*prog_name: str*) → `ArgumentParser`

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → `None`

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.AdoptBaremetalNode(app: _app.App,
                                                            app_args: Namespace |
                                                            None, cmd_name: str |
                                                            None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to adopt

PROVISION_STATE: `str` = `'adopt'`

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.AdoptBaremetalNode (WARNING)>`

```
class ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode(app: App,
                                                                        app_args:
                                                                        Namespace
                                                                        | None,
                                                                        cmd_name:
                                                                        str | None
                                                                        = None)
```

Bases: [ShowOne](#)

Show a specific BIOS setting for a node.

get_parser(*prog_name: str*) → `ArgumentParser`

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → `tuple[tuple[str, ...], tuple[Any, ...]]`

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.BootdeviceSetBaremetalNode(app:
    _app.App,
    app_args:
    Namespace |
    None,
    cmd_name:
    str | None =
    None)
```

Bases: [Command](#)

Set the boot device for a node

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.BootdeviceSetBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.BootdeviceShowBaremetalNode(app: App,
    app_args:
    Namespace |
    None,
    cmd_name:
    str | None =
    None)
```

Bases: [ShowOne](#)

Show the boot device information for a node

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.BootdeviceShowBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.BootmodeSetBaremetalNode(app: _app.App,
    app_args:
    Namespace |
    None,
    cmd_name: str |
    None = None)
```

Bases: [Command](#)

Set the boot mode for the next baremetal node deployment

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.BootmodeSetBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.CleanBaremetalNode(app: _app.App,  
                                                         app_args: Namespace |  
                                                         None, cmd_name: str |  
                                                         None = None)
```

Bases: `ProvisionStateWithWait`

Set provision state of baremetal node to clean

PROVISION_STATE: `str` = `'clean'`

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.CleanBaremetalNode (WARNING)>`

```
class ironicclient.osc.v1.baremetal_node.ConsoleDisableBaremetalNode(app:  
                                                                    _app.App,  
                                                                    app_args:  
                                                                    Namespace |  
                                                                    None,  
                                                                    cmd_name:  
                                                                    str | None =  
                                                                    None)
```

Bases: `Command`

Disable console access for a node

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.ConsoleDisableBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ConsoleEnableBaremetalNode(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Command](#)

Enable console access for a node

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.ConsoleEnableBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ConsoleShowBaremetalNode(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str |
                                                                    None = None)
```

Bases: [ShowOne](#)

Show console information for a node

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.ConsoleShowBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.CreateBaremetalNode(app: App, app_args:
                                                                    Namespace | None,
                                                                    cmd_name: str | None
                                                                    = None)
```

Bases: [ShowOne](#)

Register a new node with the baremetal service

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.CreateBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
    Run command.
```

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.DeleteBaremetalNode(app: _app.App,
                                                             app_args: Namespace
                                                             | None, cmd_name: str
                                                             | None = None)
```

Bases: [Command](#)

Unregister baremetal node(s)

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.DeleteBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.
```

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.DeployBaremetalNode(app: _app.App,
                                                             app_args: Namespace
                                                             | None, cmd_name: str
                                                             | None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to deploy

```
PROVISION_STATE: str = 'active'
```

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.DeployBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.InjectNmiBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Command](#)

Inject NMI to baremetal node

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.InjectNmiBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.InspectBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: *ProvisionStateWithWait*

Set provision state of baremetal node to inspect

```
PROVISION_STATE: str = 'inspect'
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.InspectBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode(app: App,
                                                                app_args:
                                                                Namespace
                                                                | None,
                                                                cmd_name:
                                                                str | None
                                                                = None)
```

Bases: *Lister*

List a nodes BIOS settings.

```
get_parser(prog_name: str) → ArgumentParser
```

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
```

Run command.

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, app_args:
                                                                Namespace | None,
                                                                cmd_name: str | None =
                                                                None)
```

Bases: *Lister*

List baremetal nodes

```
PROVISION_STATES: list[str] = ['active', 'deleted', 'rebuild', 'inspect',
                                'provide', 'manage', 'clean', 'adopt', 'abort']
```

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.ListBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.ListFirmwareComponentBaremetalNode(app:
                                                                           App,
                                                                           app_args:
                                                                           Names-
                                                                           pace
                                                                           |
                                                                           None,
                                                                           cmd_name:
                                                                           str
                                                                           |
                                                                           None
                                                                           =
                                                                           None)
```

Bases: `Lister`

List all Firmware Components of a node

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.ListFirmwareComponentBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

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,
                                                                  app_args:
                                                                  Namespace |
                                                                  None, cmd_name:
                                                                  str | None =
                                                                  None)
```

Bases: `Lister`

List a nodes traits.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.ListTraitsBaremetalNode (WARNING)>`

take_action(*parsed_args: Namespace*) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Command](#)

Set baremetal node to maintenance mode

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.MaintenanceSetBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Names-
                                                                    pace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None
                                                                    = None)
```

Bases: [Command](#)

Unset baremetal node from maintenance mode

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ManageBaremetalNode(app: _app.App,
                                                                app_args: Namespace
                                                                | None, cmd_name: str
                                                                | None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to manage

PROVISION_STATE: str = 'manage'

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.ManageBaremetalNode (WARNING)>

```
class ironicclient.osc.v1.baremetal_node.NodeChildrenList(app: App, app_args:
    Namespace | None,
    cmd_name: str | None =
    None)
```

Bases: [ShowOne](#)

Get a list of nodes associated as children.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.NodeChildrenList (WARNING)>

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.NodeHistoryEventGet(app: App, app_args:
    Namespace | None,
    cmd_name: str | None
    = None)
```

Bases: [ShowOne](#)

Get history event for a baremetal node.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.NodeHistoryEventGet (WARNING)>

take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.NodeHistoryList(app: App, app_args:
    Namespace | None,
    cmd_name: str | None =
    None)
```

Bases: [Lister](#)

Get history events for a baremetal node.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.NodeHistoryList` (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.NodeInventorySave(app: _app.App,  
                                                         app_args: Namespace |  
                                                         None, cmd_name: str |  
                                                         None = None)
```

Bases: `Command`

Get hardware inventory of a node (in JSON format) or save it to file.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.NodeInventorySave` (WARNING)>

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.PassthruCallBaremetalNode(app:  
                                                                    _app.App,  
                                                                    app_args:  
                                                                    Namespace |  
                                                                    None,  
                                                                    cmd_name: str  
                                                                    | None =  
                                                                    None)
```

Bases: `Command`

Call a vendor passthru method for a node

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_node.PassthruCallBaremetalNode` (WARNING)>

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.PassthruListBaremetalNode(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: [Lister](#)

List vendor passthru methods for a node

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.PassthruListBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App,
                                                             app_args: Namespace |
                                                             None, cmd_name: str |
                                                             None = None)
```

Bases: [Command](#)

Base power state class, for setting the power of a node

POWER_STATE: str

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.PowerBaremetalNode (WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.PowerOffBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [PowerBaremetalNode](#)

Power off a node

POWER_STATE: str = 'off'

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.PowerOffBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.PowerOnBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [PowerBaremetalNode](#)

Power on a node

POWER_STATE: str = 'on'

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.PowerOnBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.ProvideBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to provide

PROVISION_STATE: str = 'provide'

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ProvideBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode(app:
                                                                _app.App,
                                                                app_args:
                                                                Namespace |
                                                                None,
                                                                cmd_name:
                                                                str | None =
                                                                None)
```

Bases: [Command](#)

Base provision state class

PROVISION_STATE: str

get_parser(prog_name: str) → ArgumentParser
Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode (WARNING)>
```

take_action(parsed_args: Namespace) → None
Override to do something useful.
The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [ProvisionStateBaremetalNode](#)

Provision state class adding wait flag.

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.

    The returned value will be returned by the program.
```

```
class ironicclient.osc.v1.baremetal_node.RebootBaremetalNode(app: _app.App,
                                                             app_args: Namespace
                                                             | None, cmd_name: str
                                                             | None = None)
```

Bases: [Command](#)

Reboot baremetal node

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.RebootBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.

    The returned value will be returned by the program.
```

```
class ironicclient.osc.v1.baremetal_node.RebuildBaremetalNode(app: _app.App,
                                                             app_args:
                                                             Namespace | None,
                                                             cmd_name: str |
                                                             None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to rebuild

```
PROVISION_STATE: str = 'rebuild'
```

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.RebuildBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.RemoveTraitBaremetalNode(app: _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str |
                                                                    None = None)
```

Bases: [Command](#)

Remove trait(s) from a node.

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.RemoveTraitBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.

    The returned value will be returned by the program.
```

```
class ironicclient.osc.v1.baremetal_node.RescueBaremetalNode(app: _app.App,
                                                              app_args: Namespace
                                                              | None, cmd_name: str
                                                              | None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to rescue

```
PROVISION_STATE: str = 'rescue'
```

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.RescueBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.SecurebootOffBaremetalNode(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Command](#)

Turn secure boot off

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.SecurebootOffBaremetalNode (WARNING)>
```

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.SecurebootOnBaremetalNode(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: [Command](#)

Turn secure boot on

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.SecurebootOnBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ServiceBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to service

PROVISION_STATE: str = 'service'

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.ServiceBaremetalNode (WARNING)>

```
class ironicclient.osc.v1.baremetal_node.SetBaremetalNode(app: _app.App, app_args:
                                                            Namespace | None,
                                                            cmd_name: str | None =
                                                            None)
```

Bases: [Command](#)

Set baremetal properties

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.SetBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ShowBaremetalNode(app: App, app_args:
                                                                Namespace | None,
                                                                cmd_name: str | None =
                                                                None)
```

Bases: [ShowOne](#)

Show baremetal node details

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ShowBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
```

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_node.UndeployBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [ProvisionStateWithWait](#)

Set provision state of baremetal node to deleted

```
PROVISION_STATE: str = 'deleted'
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.UndeployBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.UnholdBaremetalNode(app: _app.App,
                                                                app_args: Namespace
                                                                | None, cmd_name: str
                                                                | None = None)
```

Bases: [ProvisionStateBaremetalNode](#)

Set provision state of baremetal node to unhold

```
PROVISION_STATE: str = 'unhold'
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.UnholdBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.UnrescueBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: *ProvisionStateWithWait*

Set provision state of baremetal node to unrescue

PROVISION_STATE: str = 'unrescue'

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.UnrescueBaremetalNode (WARNING)>
```

```
class ironicclient.osc.v1.baremetal_node.UnsetBaremetalNode(app: _app.App,
                                                            app_args: Namespace |
                                                            None, cmd_name: str |
                                                            None = None)
```

Bases: *Command*

Unset baremetal properties

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.UnsetBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.
```

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.ValidateBaremetalNode(app: App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: *Lister*

Validate a nodes driver interfaces

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_node.ValidateBaremetalNode (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
    Run command.
```

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.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Command](#)

Attach VIF to a given node

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.VifAttachBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.VifDetachBaremetalNode(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Command](#)

Detach VIF from a given node

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.VifDetachBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_node.VifListBaremetalNode(app: App, app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Lister](#)

Show attached VIFs for a node

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_node.VifListBaremetalNode (WARNING)>

take_action(*parsed_args: Namespace*) → tuple[Sequence[str], Iterable[Any]]

Run command.

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, app_args:
    Namespace | None,
    cmd_name: str | None
    = None)
```

Bases: [ShowOne](#)

Create a new port

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_port.CreateBaremetalPort (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
    Run command.

    Return a tuple containing the column names and an iterable containing the data to be listed.
```

```
class ironicclient.osc.v1.baremetal_port.DeleteBaremetalPort(app: _app.App,
    app_args: Namespace
    | None, cmd_name: str
    | None = None)
```

Bases: [Command](#)

Delete port(s).

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_port.DeleteBaremetalPort (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
    Override to do something useful.

    The returned value will be returned by the program.
```

```
class ironicclient.osc.v1.baremetal_port.ListBaremetalPort(app: App, app_args:
    Namespace | None,
    cmd_name: str | None =
    None)
```

Bases: [Lister](#)

List baremetal ports.

```
get_parser(prog_name: str) → ArgumentParser
    Return an argparse.ArgumentParser.
```

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_port.ListBaremetalPort (WARNING)>
```

take_action(*parsed_args: Namespace*) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App, app_args:
    Namespace | None,
    cmd_name: str | None =
    None)
```

Bases: [Command](#)

Set baremetal port properties.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_port.SetBaremetalPort (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_port.ShowBaremetalPort(app: App, app_args:
    Namespace | None,
    cmd_name: str | None =
    None)
```

Bases: [ShowOne](#)

Show baremetal port details.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_port.ShowBaremetalPort (WARNING)>

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_port.UnsetBaremetalPort(app: _app.App,
    app_args: Namespace |
    None, cmd_name: str |
    None = None)
```

Bases: [Command](#)

Unset baremetal port properties.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_port.UnsetBaremetalPort (WARNING)>

take_action(*parsed_args: Namespace*) → None

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,  
                                                                    app_args:  
                                                                    Names-  
                                                                    pace |  
                                                                    None,  
                                                                    cmd_name:  
                                                                    str | None  
                                                                    = None)
```

Bases: [ShowOne](#)

Create a new baremetal port group.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup
(WARNING)>

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup(app:  
                                                                    _app.App,  
                                                                    app_args:  
                                                                    Names-  
                                                                    pace |  
                                                                    None,  
                                                                    cmd_name:  
                                                                    str | None  
                                                                    = None)
```

Bases: [Command](#)

Unregister baremetal port group(s).

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup
(WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_portgroup.ListBaremetalPortGroup(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Lister](#)

List baremetal port groups.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_portgroup.ListBaremetalPortGroup (WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [Command](#)

Set baremetal port group properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_portgroup.SetBaremetalPortGroup (WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: [ShowOne](#)

Show baremetal port group details.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace
                                                                    | None,
                                                                    cmd_name:
                                                                    str | None
                                                                    = None)
```

Bases: `Command`

Unset baremetal port group properties.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

ironicclient.osc.v1.baremetal_runbook module

```
class ironicclient.osc.v1.baremetal_runbook.AddTraitBaremetalRunbook(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: `Command`

Add traits to a runbook.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log = `<Logger`

`ironicclient.osc.v1.baremetal_runbook.AddTraitBaremetalRunbook (WARNING)>`

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_runbook.CreateBaremetalRunbook(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: [ShowOne](#)

Create a new runbook

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_runbook.CreateBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_runbook.DeleteBaremetalRunbook(app:
                                                                    _app.App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None,
                                                                    cmd_name: str
                                                                    | None =
                                                                    None)
```

Bases: [Command](#)

Delete runbook(s).

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_runbook.DeleteBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_runbook.ListBaremetalRunbook(app: App,
                                                                    app_args:
                                                                    Namespace |
                                                                    None, cmd_name:
                                                                    str | None =
                                                                    None)
```

Bases: *Lister*

List baremetal runbooks.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an *argparse.ArgumentParser*.

log: *logging.Logger* = *<Logger*

ironicclient.osc.v1.baremetal_runbook.ListBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_runbook.ListTraitsBaremetalRunbook(app: App,  
                                                                    app_args:  
                                                                    Names-  
                                                                    pace |  
                                                                    None,  
                                                                    cmd_name:  
                                                                    str | None  
                                                                    = None)
```

Bases: *Lister*

List a runbooks traits.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an *argparse.ArgumentParser*.

log = *<Logger*

ironicclient.osc.v1.baremetal_runbook.ListTraitsBaremetalRunbook
(WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], list[list[str]]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_runbook.RemoveTraitBaremetalRunbook(app:  
                                                                    _app.App,  
                                                                    app_args:  
                                                                    Names-  
                                                                    pace |  
                                                                    None,  
                                                                    cmd_name:  
                                                                    str |  
                                                                    None =  
                                                                    None)
```

Bases: *Command*

Remove trait(s) from a runbook.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an *argparse.ArgumentParser*.

```
log = <Logger
ironicclient.osc.v1.baremetal_runbook.RemoveTraitBaremetalRunbook
(WARNING)>
```

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_runbook.SetBaremetalRunbook(app: _app.App,
                                                                app_args:
                                                                Namespace | None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: [Command](#)

Set baremetal runbook properties.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_runbook.SetBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_runbook.ShowBaremetalRunbook(app: App,
                                                                app_args:
                                                                Namespace |
                                                                None, cmd_name:
                                                                str | None =
                                                                None)
```

Bases: [ShowOne](#)

Show baremetal runbook details.

get_parser(*prog_name: str*) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_runbook.ShowBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → tuple[tuple[str, ...], tuple[Any, ...]]

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_runbook.UnsetBaremetalRunbook(app: _app.App,
                                                                app_args:
                                                                Namespace |
                                                                None,
                                                                cmd_name: str |
                                                                None = None)
```

Bases: *Command*

Unset baremetal runbook properties.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an *argparse.ArgumentParser*.

log: *logging.Logger* = *<Logger*

ironicclient.osc.v1.baremetal_runbook.UnsetBaremetalRunbook (WARNING)>

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

ironicclient.osc.v1.baremetal_shard module

```
class ironicclient.osc.v1.baremetal_shard.ListBaremetalShard(app: App, app_args: Namespace | None, cmd_name: str | None = None)
```

Bases: *Lister*

List baremetal shards.

log: *logging.Logger* = *<Logger*

ironicclient.osc.v1.baremetal_shard.ListBaremetalShard (WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[Sequence[str], Iterable[Any]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

ironicclient.osc.v1.baremetal_volume_connector module

```
class ironicclient.osc.v1.baremetal_volume_connector.CreateBaremetalVolumeConnector(app: App, app_args: Namespace | None, cmd_name: str | None = None)
```

Bases: *ShowOne*

Create a new baremetal volume connector.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an *argparse.ArgumentParser*.

```
log: logging.Logger = <Logger ironicclient.osc.v1.
baremetal_volume_connector.CreateBaremetalVolumeConnector (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[tuple[str, ...], tuple[Any, ...]]
```

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_volume_connector.DeleteBaremetalVolumeConnector(app:
    _app.App,
    app_args: argparse.Namespace,
    cmd_name: str) =
    None)
```

Bases: [Command](#)

Unregister baremetal volume connector(s).

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger ironicclient.osc.v1.
baremetal_volume_connector.DeleteBaremetalVolumeConnector (WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_volume_connector.ListBaremetalVolumeConnector(app:
    App,
    app_args: argparse.Namespace,
    cmd_name: str) =
    None)
```

Bases: [Lister](#)

List baremetal volume connectors.

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger ironicclient.osc.v1.
baremetal_volume_connector.ListBaremetalVolumeConnector (WARNING)>
```

```
take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]
```

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_volume_connector.SetBaremetalVolumeConnector(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [Command](#)

Set baremetal volume connector properties.

```
get_parser(prog_name: str) → ArgumentParser
```

Return an `argparse.ArgumentParser`.

```
log: logging.Logger = <Logger
ironicclient.osc.v1.baremetal_volume_connector.SetBaremetalVolumeConnector
(WARNING)>
```

```
take_action(parsed_args: Namespace) → None
```

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_volume_connector.ShowBaremetalVolumeConnector(app:
    App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [ShowOne](#)

Show baremetal volume connector details.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger ironicclient.osc.v1.`

`baremetal_volume_connector.ShowBaremetalVolumeConnector (WARNING)>`

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_volume_connector.UnsetBaremetalVolumeConnector(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name
    str
    |
    None
    =
    None)
```

Bases: [Command](#)

Unset baremetal volume connector properties.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger ironicclient.osc.v1.`

`baremetal_volume_connector.UnsetBaremetalVolumeConnector (WARNING)>`

take_action(*parsed_args: Namespace*) → *None*

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,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [ShowOne](#)

Create a new baremetal volume target.

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget`
(WARNING)>

take_action(*parsed_args: Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget(app:  
    _app.App,  
    app_args:  
    Names-  
    pace  
    |  
    None,  
    cmd_name:  
    str  
    |  
    None  
    =  
    None)
```

Bases: [Command](#)

Unregister baremetal volume target(s).

get_parser(*prog_name: str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget`
(WARNING)>

take_action(*parsed_args: Namespace*) → *None*

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget(app:
                                                                    App,
                                                                    app_args:
                                                                    Names-
                                                                    pace
                                                                    |
                                                                    None,
                                                                    cmd_name:
                                                                    str
                                                                    |
                                                                    None
                                                                    =
                                                                    None)
```

Bases: [Lister](#)

List baremetal volume targets.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget

(WARNING)>

take_action(parsed_args: Namespace) → tuple[Sequence[str], Iterable[Any]]

Run command.

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.App,
                                                                    app_args:
                                                                    Names-
                                                                    pace
                                                                    |
                                                                    None,
                                                                    cmd_name:
                                                                    str |
                                                                    None
                                                                    =
                                                                    None)
```

Bases: [Command](#)

Set baremetal volume target properties.

get_parser(prog_name: str) → ArgumentParser

Return an argparse.ArgumentParser.

log: logging.Logger = <Logger

ironicclient.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget

(WARNING)>

take_action(parsed_args: Namespace) → None

Override to do something useful.

The returned value will be returned by the program.

```
class ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget(app:
    App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [*ShowOne*](#)

Show baremetal volume target details.

get_parser(*prog_name*: *str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget`
(WARNING)>

take_action(*parsed_args*: *Namespace*) → *tuple[tuple[str, ...], tuple[Any, ...]]*

Run command.

Return a tuple containing the column names and an iterable containing the data to be listed.

```
class ironicclient.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget(app:
    _app.App,
    app_args:
    Names-
    pace
    |
    None,
    cmd_name:
    str
    |
    None
    =
    None)
```

Bases: [*Command*](#)

Unset baremetal volume target properties.

get_parser(*prog_name*: *str*) → *ArgumentParser*

Return an `argparse.ArgumentParser`.

log: `logging.Logger` = `<Logger`

`ironicclient.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget`
(WARNING)>

take_action(*parsed_args: Namespace*) → None

Override to do something useful.

The returned value will be returned by the program.

Module contents

Submodules

ironicclient.osc.command module

Typed base command classes for ironicclient OSC commands.

These subclass osc-libs command classes and annotate `app` so that `self.app.client_manager.baremetal` is statically known to be `ironicclient.v1.client.Client`, removing the need for `# type: ignore[attr-defined]` throughout the command implementations.

See <https://review.opendev.org/c/openstack/python-openstackclient/+/-/970681> for the equivalent pattern in python-openstackclient.

```
class ironicclient.osc.command.Command(app: _app.App, app_args: Namespace | None,  
                                       cmd_name: str | None = None)
```

Bases: `Command`

app: `_App`

log = `<Logger ironicclient.osc.command.Command (WARNING)>`

```
class ironicclient.osc.command.Lister(app: App, app_args: Namespace | None, cmd_name:  
                                     str | None = None)
```

Bases: `Command`, `Lister`

log = `<Logger ironicclient.osc.command.Lister (WARNING)>`

```
class ironicclient.osc.command.ShowOne(app: App, app_args: Namespace | None,  
                                       cmd_name: str | None = None)
```

Bases: `Command`, `ShowOne`

log = `<Logger ironicclient.osc.command.ShowOne (WARNING)>`

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: `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: ArgumentParser) → ArgumentParser`

Hook to add global options.

`ironicclient.osc.plugin.make_client(instance: ClientManager) → Client`

Returns a baremetal service client.

Module contents

ironicclient.v1 package

Submodules

ironicclient.v1.allocation module

`class ironicclient.v1.allocation.Allocation(manager: ManagerProtocol, info: dict[str, Any], loaded: bool = False)`

Bases: *Resource*

`class ironicclient.v1.allocation.AllocationManager(api: SessionClient)`

Bases: *CreateManager*[*Allocation*]

`delete(allocation_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → 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: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → Resource | 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*: str | None = None, *state*: str | None = None, *node*: str | None = None, *limit*: int | None = None, *marker*: str | None = None, *sort_key*: str | None = None, *sort_dir*: str | None = None, *fields*: list[str] | None = None, *owner*: str | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[[Allocation](#)]

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*: str, *patch*: list[dict[str, Any]], *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Resource](#)

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*: str, *timeout*: int | float = 0, *poll_interval*: int | float = 1, *poll_delay_function*: Callable[[int | float], object] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Resource](#) | 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*: [ManagerProtocol](#), *info*: dict[str, Any], *loaded*: bool = False)

Bases: [Resource](#)

class `ironicclient.v1.chassis.ChassisManager`(*api*: [SessionClient](#))

Bases: [CreateManager\[Chassis\]](#)

delete(*chassis_id*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → None

get(*chassis_id*: str, *fields*: list[str] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Resource](#) | None

list(*marker*: str | None = None, *limit*: int | None = None, *sort_key*: str | None = None, *sort_dir*: str | None = None, *detail*: bool = False, *fields*: list[str] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[[Chassis](#)]

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: str, marker: str | None = None, limit: int | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, associated: bool | str | None = None, maintenance: bool | str | None = None, provision_state: str | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → list[[Chassis](#)]

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*: str, *patch*: list[dict[str, Any]], *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [*Resource*](#)

`ironicclient.v1.client` module

`class ironicclient.v1.client.Client`(*endpoint_override*: str | None = None, **args*: Any, ***kwargs*: Any)

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: `str | list[str]`

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: `bool`

Returns True if microversion negotiation has occurred.

negotiate_api_version() $\rightarrow$ `str`

Triggers negotiation with the remote API endpoint.

Returns

the negotiated API version.

ironicclient.v1.conductor module

class `ironicclient.v1.conductor.Conductor`(*manager*: [ManagerProtocol](#), *info*: `dict[str, Any]`,
loaded: `bool = False`)

Bases: [Resource](#)

class `ironicclient.v1.conductor.ConductorManager`(*api*: [SessionClient](#))

Bases: [Manager](#)[[Conductor](#)]

get(*hostname*: `str`, *fields*: `list[str] | None = None`, *os_ironic_api_version*: `str | None = None`,
global_request_id: `str | None = None`) $\rightarrow$ [Resource](#) | `None`

list(*marker*: `str | None = None`, *limit*: `int | None = None`, *sort_key*: `str | None = None`,
sort_dir: `str | None = None`, *fields*: `list[str] | None = None`, *detail*: `bool = False`,
os_ironic_api_version: `str | None = None`, *global_request_id*: `str | None = None`) $\rightarrow$
`list`[[Conductor](#)]

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: Client, chassis_list: list[dict[str, object]]) → list[Exception]`

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: Client, node_list: list[dict[str, object]], chassis_uuid: str | None = None) → list[Exception]`

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: Client, portgroup_list: list[dict[str, object]], node_uuid: str) → list[Exception]`

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: Client, port_list: list[dict[str,
                                                                    object]], node_uuid: str, portgroup_uuid:
                                                                    str | None = None) → list[Exception]
```

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: Client, filenames: list[str]) →
                                                                    None
```

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: Client, **params:
                                                                    Any) → str
```

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: str) →
                                                                    Callable[[Callable[[...], str]],
                                                                    Callable[[...], tuple[str | None,
                                                                    Exception | None]]]
```

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: Client, **params: Any) → str`

Call the client to create a node.

Parameters

- **client** ironic client instance.
- **params** dictionary to be POSTed to `/nodes` endpoint, excluding ports and port-groups 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: Client, **params: Any) → str`

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: Client, **params: Any) → str`

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: str) → dict[str, object]`

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: ManagerProtocol, info: dict[str, Any], loaded: bool = False)
```

Bases: [Resource](#)

```
class ironicclient.v1.deploy_template.DeployTemplateManager(api: SessionClient)
```

Bases: [CreateManager\[DeployTemplate\]](#)

```
delete(template_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → None
```

```
get(template_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → Resource | None
```

```
list(limit: int | None = None, marker: str | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → list[DeployTemplate]
```

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: str, patch: list[dict[str, Any]], os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [`Resource`](#)

ironicclient.v1.driver module

class `ironicclient.v1.driver.Driver`(*manager: [ManagerProtocol](#), info: dict[str, Any], loaded: bool = False*)

Bases: [`Resource`](#)

class `ironicclient.v1.driver.DriverManager`(*api: [SessionClient](#)*)

Bases: [`Manager\[Driver\]`](#)

delete(*driver_name: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → `None`

get(*driver_name: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None, fields: list[str] | None = None*) → [`Resource`](#) | `None`

get_vendor_passthru_methods(*driver_name: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, Any]

list(*driver_type: str | None = None, detail: bool | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None, fields: list[str] | None = None*) → list[[`Driver`](#)]

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.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.

Returns

A list of drivers.

properties(*driver_name: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, str]

raid_logical_disk_properties(*driver_name: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, str]

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: str, patch: list[dict[str, Any]], http_method: str = 'PATCH', os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Resource](#)

vendor_passthru(*driver_name: str, method: str, args: dict[str, Any] | None = None, http_method: str | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Resource](#) | 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: [ManagerProtocol](#), info: dict[str, Any], loaded: bool = False)

Bases: [Resource](#)

class ironicclient.v1.events.**EventManager**(api: [SessionClient](#))

Bases: [CreateManager\[Event\]](#)

resource_class

alias of [Event](#)

ironicclient.v1.inspection_rule module

class ironicclient.v1.inspection_rule.**InspectionRule**(manager: [ManagerProtocol](#), info: dict[str, Any], loaded: bool = False)

Bases: [Resource](#)

class ironicclient.v1.inspection_rule.**InspectionRuleManager**(api: [SessionClient](#))

Bases: [CreateManager\[InspectionRule\]](#)

delete(rule_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → None

get(rule_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → [Resource](#) | None

list(limit: int | None = None, marker: str | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → list[[InspectionRule](#)]

Retrieve a list of rules.

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 rules to return.

- 2) `limit == 0`, return the entire list of rules.
- 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 rules.
- **`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 rules.

`resource_class`

alias of [*InspectionRule*](#)

`update`(*rule_id*: str, *patch*: list[dict[str, Any]], *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [*Resource*](#)

ironicclient.v1.node module

`class ironicclient.v1.node.Node`(*manager*: [*ManagerProtocol*](#), *info*: dict[str, Any], *loaded*: bool = False)

Bases: [*Resource*](#)

`class ironicclient.v1.node.NodeManager`(*api*: [*SessionClient*](#))

Bases: [*CreateManager\[Node\]*](#)

`add_trait`(*node_ident*: str, *trait*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [*Node*](#) | 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*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → None

get(*node_id*: str, *fields*: list[str] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Node](#) | None

get_bios_setting(*node_ident*: str, *name*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → dict[str, str] | 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*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → dict[str, str | bool]

get_by_instance_uuid(*instance_uuid*: str, *fields*: list[str] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Node](#)

get_console(*node_uuid*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → dict[str, bool | dict[str, str] | None]

get_history_event(*node_ident*: str, *event*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → dict[str, str]

Get a single event record for a node.

Provides the ability to request, and return a nodes single event history entry.

Parameters

- **node_ident** The name or UUID of the node.
- **event** The UUID of the event entry as listed in the node event history list.
- **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_history_list(*node_ident*: str, *detail*: bool = False, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[dict[str, str]]

Get node history event list.

Provides the ability to query a node event history list from the API and return the API response to the caller.

Requires API version 1.78.

Parameters

- **node_ident** The name or UUID of the node.

- **detail** If detailed data should be returned in the event list entry. Default False.
- **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_inventory(*node_ident: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, Any]

Get the hardware inventory of the node.

Requires API version 1.81.

Parameters

- **node_ident** The name or UUID of the node.
- **os_ironic_api_version** String version (e.g. 1.81) 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_supported_boot_devices(*node_uuid: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, list[str]]

get_traits(*node_ident: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → list[str]

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: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → dict[str, dict[str, bool | str | list[str]]]

inject_rmi(*node_uuid: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Node](#) | None

list(*associated: bool | str | None = None, maintenance: bool | str | None = None, marker: str | None = None, limit: int | None = None, detail: bool = False, sort_key: str | None = None, sort_dir: str | None = None, fields: list[str] | None = None, provision_state: str | None = None, driver: str | None = None, resource_class: str | None = None, chassis: str | None = None, fault: str | None = None, os_ironic_api_version: str | None = None, conductor_group: str | None = None, conductor: str | None = None, owner: str | None = None, retired: bool | str | None = None, lessee: str | None = None, shards: list[str] | None = None, sharded: bool | str | None = None, parent_node: str | None = None, include_children: bool | None = None, description_contains: str | None = None, global_request_id: str | None = None, instance_name: str | None = None*) → list[[Node](#)]

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.
- **shards** Optional. A list with a specified set of shards to limit node returns to.
- **sharded** Optional. Boolean value, when true get only nodes with a non-null node.shard value, when false get only nodes with a null node.shard value. None is a noop. with a non-null node.shard value.
- **parent_node** Optional. String value used to retrieve child nodes with the supplied parent node.
- **include_children** Optional. Boolean Value, only True is valid. Tells the ironic API to enumerate all child nodes which are normally hidden from the node list.
- **description_contains** Optional. String value to get nodes with description contains specified value.
- **instance_name** Optional. String value to get only nodes with the given instance_name set.

Returns

A list of nodes.

list_bios_settings(*node_ident*: str, *detail*: bool = False, *fields*: list[str] | None = None, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[dict[str, str | int | bool | list[str] | 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.
- **detail** Optional, boolean whether to return detailed information about bios settings.
- **fields** Optional, a list with a specified set of fields of the resource to be returned. Can not be used when detail is set.

list_children_of_node(*node_id*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[str]

Get a list of child nodes for the supplied node_id.

Parameters

- **node_id** The name or UUID of a 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.

Returns

A list of UUIDs representing child nodes for the supplied node_id..

list_firmware_components(node_ident: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → list[dict[str, str]]

List all firmware components 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: str, marker: str | None = None, limit: int | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → list[Port]

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: str, marker: str | None = None, limit: int | None = None,
                        sort_key: str | None = None, sort_dir: str | None = None, detail:
                        bool = False, fields: list[str] | None = None,
                        os_ironic_api_version: str | None = None, global_request_id: str |
                        None = None) → list[VolumeConnector]
```

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: str, marker: str | None = None, limit: int | None = None,
                    sort_key: str | None = None, sort_dir: str | None = None, detail: bool
                    = False, fields: list[str] | None = None, os_ironic_api_version: str |
                    None = None, global_request_id: str | None = None) →
                    list[VolumeTarget]
```

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_id*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → None

Remove all traits from a node.

Parameters

- **node_id** 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_id*: str, *trait*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → 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: str, boot_device: str, persistent: bool = False, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Node](#) | None

set_boot_mode(*node_id: str, state: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Node](#) | None

Sets boot mode for a node.

Parameters

- **node_id** Node identifier
- **state** One of target boot modes, uefi or bios
- **os_ironic_api_version** String version (e.g. 1.76) 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 boot mode is not one of uefi / bios

Returns

The status of the request

set_console_mode(*node_uuid: str, enabled: bool | str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Node](#) | 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: str, state: bool | str, maint_reason: str | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Node](#) | 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: str, state: str, soft: bool = False, timeout: int | None = None,
                os_ironic_api_version: str | None = None, global_request_id: str | None =
                None) → Node | 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: str, state: str, configdrive: str | dict[str, Any] | list[Any] |
                    bytes | None = None, cleansteps: list[dict[str, Any]] | None = None,
                    rescue_password: str | None = None, os_ironic_api_version: str |
                    None = None, global_request_id: str | None = None, deploysteps:
                    list[dict[str, Any]] | None = None, servicesteps: list[dict[str, Any]] |
                    None = None, runbook: str | None = None, disable_ramdisk: bool | str
                    | None = None) → Node | 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** One of:
 - a gzipped, base64-encoded configuration drive string
 - a dictionary to build config drive from
 - a path to the configuration drive file (ISO 9660 or VFAT)
 - a path to a directory containing the config drive files
 - a path to a JSON file to build config from

In case its a directory, a config drive will be generated from it. In case its a dictionary or a JSON file, 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.
- **deploysteps** The deploy steps as a list of deploy-step dictionaries; each dictionary should have keys interface, step, priority, and optional key args. This is optional and is only valid when setting provision-state to active or rebuild.
- **servicesteps** The service steps as list of service-step dictionaries; each dictionary should have keys interface, step, and optional key args when setting an active nodes to service.
- **runbook** The identifier of a predefined runbook to use for provisioning.
- **disable_ramdisk** Boolean if set to true will not boot the ironic-python-agent for cleaning. Only valid when setting state to clean or service and only for steps explicitly marked as not requiring the ironic-python-agent can use this.

Raises

InvalidAttribute if there was an error with the clean steps or deploy steps

Returns

The status of the request

set_secure_boot(*node_id*: str, *state*: bool | str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → *Node* | None

Set the secure boot state for the node.

Parameters

- **node_id** The UUID of the node.
- **state** the secure boot state; either a Boolean or a string representation of a Boolean (eg, true, on, false, off). True to turn secure boot on; False to turn secure boot off.
- **os_ironic_api_version** String version (e.g. 1.76) 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_target_raid_config(node_ident: str, target_raid_config: dict[str, Any],  
                       os_ironic_api_version: str | None = None, global_request_id: str |  
                       None = None) → Node | 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: str, traits: list[str], os_ironic_api_version: str | None = None,  
           global_request_id: str | None = None) → Node | 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: str, os_ironic_api_version: str | None = None, global_request_id: str |  
       None = None) → Node | None
```

```
update(node_id: str, patch: list[dict[str, Any]] | dict[str, Any] | None, http_method: str =  
       'PATCH', os_ironic_api_version: str | None = None, reset_interfaces: bool | None =  
       None, global_request_id: str | None = None) → Node | None
```

validate(*node_uuid: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → *Node* | *None*

vendor_passthru(*node_id: str, method: str, args: dict[str, Any] | None = None, http_method: str | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → *Node* | *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: str, vif_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None, **kwargs: str*) → *None*

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: str, vif_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → *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*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[Node]

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*: str | list[str], *expected_state*: str, *timeout*: int | float = 0, *poll_interval*: int | float = 2, *poll_delay_function*: Callable[[int | float], object] | None = None, *fail_on_unexpected_state*: bool = True, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → None

Helper function to wait for nodes 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 (one or a list)
- **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: ManagerProtocol, info: dict[str, Any], loaded:
                                bool = False)
```

Bases: [Resource](#)

```
class ironicclient.v1.port.PortManager(api: SessionClient)
```

Bases: [CreateManager\[Port\]](#)

```
delete(port_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None =
        None) → None
```

```
get(port_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None,
    global_request_id: str | None = None) → Port | None
```

```
get_by_address(address: str, fields: list[str] | None = None, os_ironic_api_version: str |
               None = None, global_request_id: str | None = None) → Port
```

```
list(address: str | None = None, limit: int | None = None, marker: str | None = None, sort_key:
      str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None
      = None, node: str | None = None, portgroup: str | None = None, os_ironic_api_version:
      str | None = None, global_request_id: str | None = None) → list[Port]
```

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: str, patch: list[dict[str, Any]], os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Port](#)

ironicclient.v1.portgroup module

class `ironicclient.v1.portgroup.Portgroup`(*manager: [ManagerProtocol](#), info: dict[str, Any], loaded: bool = False*)

Bases: [Resource](#)

class `ironicclient.v1.portgroup.PortgroupManager`(*api: [SessionClient](#)*)

Bases: [CreateManager\[Portgroup\]](#)

delete(*portgroup_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → 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: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → [Portgroup](#) | 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: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → *Portgroup*

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: str | None = None, address: str | None = None, limit: int | None = None, marker: str | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, shards: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → list[*Portgroup*]

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.
- **shards** Optional, a list of shards to filter portgroups by. Only portgroups associated with nodes in these shards will be returned.
- **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: str, marker: str | None = None, limit: int | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → list[Portgroup]

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*: str, *patch*: list[dict[str, Any]], *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → *Portgroup*

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: list[str], sort_excluded:  
                                              list[str] | None = None, override_labels:  
                                              dict[str, str] | None = 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: dict[str, str] = {'actions': 'Actions', 'address': 'Address',
'alive': 'Alive', 'allocation_uuid': 'Allocation UUID',
'allowable_values': 'Allowable Values', 'async': 'Async', 'attach':
'Response is attachment', 'attribute_type': 'Attribute Type',
'automated_clean': 'Automated Clean', 'bios_interface': 'BIOS
Interface', 'boot_index': 'Boot Index', 'boot_interface': 'Boot
Interface', 'boot_mode': 'Boot Mode', 'candidate_nodes': 'Candidate
Nodes', 'category': 'Category', 'chassis_uuid': 'Chassis UUID',
'children': 'Child Nodes', 'clean_step': 'Clean Step', 'conditions':
'Conditions', 'conductor': 'Conductor', 'conductor_group': 'Conductor
Group', 'connector_id': 'Connector ID', 'console_enabled': 'Console
Enabled', 'console_interface': 'Console Interface', 'count': 'Count',
'created_at': 'Created At', 'default_bios_interface': 'Default BIOS
Interface', 'default_boot_interface': 'Default Boot Interface',
'default_console_interface': 'Default Console Interface',
'default_deploy_interface': 'Default Deploy Interface',
'default_firmware_interface': 'Default Firmware Interface',
'default_inspect_interface': 'Default Inspect Interface',
'default_management_interface': 'Default Management Interface',
'default_network_interface': 'Default Network Interface',
'default_power_interface': 'Default Power Interface',
'default_raid_interface': 'Default RAID Interface',
'default_rescue_interface': 'Default Rescue Interface',
'default_storage_interface': 'Default Storage Interface',
'default_vendor_interface': 'Default Vendor Interface',
'deploy_interface': 'Deploy Interface', 'deploy_step': 'Deploy Step',
'description': 'Description', 'disable_power_off': 'Disable Power Off',
'driver': 'Driver', 'driver_info': 'Driver Info',
'driver_internal_info': 'Driver Internal Info', 'drivers': 'Drivers',
'enabled_bios_interfaces': 'Enabled BIOS Interfaces',
'enabled_boot_interfaces': 'Enabled Boot Interfaces',
'enabled_console_interfaces': 'Enabled Console Interfaces',
'enabled_deploy_interfaces': 'Enabled Deploy Interfaces',
'enabled_firmware_interfaces': 'Enabled Firmware Interfaces',
'enabled_inspect_interfaces': 'Enabled Inspect Interfaces',
'enabled_management_interfaces': 'Enabled Management Interfaces',
'enabled_network_interfaces': 'Enabled Network Interfaces',
'enabled_power_interfaces': 'Enabled Power Interfaces',
'enabled_raid_interfaces': 'Enabled RAID Interfaces',
'enabled_rescue_interfaces': 'Enabled Rescue Interfaces',
'enabled_storage_interfaces': 'Enabled Storage Interfaces',
'enabled_vendor_interfaces': 'Enabled Vendor Interfaces', 'event':
'Description of the event', 'event_type': 'Event Origin Type', 'extra':
'Extra', 'fault': 'Fault', 'firmware_interface': 'Firmware Interface',
'health': 'Health', 'hostname': 'Hostname', 'hosts': 'Active host(s)',
'http_methods': 'Supported HTTP methods', 'id': 'ID',
'inspect_interface': 'Inspect Interface', 'inspection_finished_at':
'Inspection Finished At', 'inspection_started_at': 'Inspection Started
At', 'instance_info': 'Instance Info', 'instance_name': 'Instance Name',
'instance_uuid': 'Instance UUID', 'internal_info': 'Internal Info',
'is_smartnic': 'Is Smart NIC port', 'last_error': 'Last Error',
'lessee': 'Lessee', 'local_link_connection': 'Local Link Connection',
'lower_bound': 'Lower Bound', 'maintenance': 'Maintenance',
'maintenance_reason': 'Maintenance Reason', 'management_interface':
'Management Interface', 'max_length': 'Maximum Length', 'min_length':
'Minimum Length', 'mode': 'Mode', 'name': 'Name', 'network_data':
'Network Configuration', 'network_interface': 'Network Interface',
```

```
property fields: tuple[str, ...]
property labels: tuple[str, ...]
property sort_fields: tuple[str, ...]
property sort_labels: tuple[str, ...]
```

ironicclient.v1.runbook module

```
class ironicclient.v1.runbook.Runbook(manager: ManagerProtocol, info: dict[str, Any],
                                       loaded: bool = False)
```

Bases: [Resource](#)

```
class ironicclient.v1.runbook.RunbookManager(api: SessionClient)
```

Bases: [CreateManager\[Runbook\]](#)

```
add_trait(runbook_ident: str, trait: str, os_ironic_api_version: str | None = None,
          global_request_id: str | None = None) → Runbook | None
```

Add a trait to a runbook.

Requires API version 1.112 or later.

Parameters

- **runbook_ident** runbook UUID or name.
- **trait** trait to add to the runbook.
- **os_ironic_api_version** String version (e.g. 1.112) 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(runbook_id: str, os_ironic_api_version: str | None = None, global_request_id: str |
        None = None) → None
```

```
get(runbook_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None,
    global_request_id: str | None = None) → Resource | None
```

```
get_traits(runbook_ident: str, os_ironic_api_version: str | None = None, global_request_id:
            str | None = None) → list[str]
```

Get traits for a runbook.

Requires API version 1.112 or later.

Parameters

- **runbook_ident** runbook UUID or name.
- **os_ironic_api_version** String version (e.g. 1.112) 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(*limit: int | None = None, marker: str | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None, project: str | None = None, public: bool | None = None*) → list[[Runbook](#)]

Retrieve a list of runbooks.

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 runbooks to return.
- 2) limit == 0, return the entire list of runbooks.
- 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 runbooks.
- **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 runbooks.

remove_all_traits(*runbook_ident: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → None

Remove all traits from a runbook.

Requires API version 1.112 or later.

Parameters

- **runbook_ident** runbook UUID or name.
- **os_ironic_api_version** String version (e.g. 1.112) 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(*runbook_ident*: str, *trait*: str, *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → None

Remove a trait from a runbook.

Requires API version 1.112 or later.

Parameters

- **runbook_ident** runbook UUID or name.
- **trait** trait to remove from the runbook.
- **os_ironic_api_version** String version (e.g. 1.112) 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 [Runbook](#)

update(*runbook_id*: str, *patch*: list[dict[str, Any]], *os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → [Resource](#)

ironicclient.v1.shard module

class `ironicclient.v1.shard.Shard`(*manager*: [ManagerProtocol](#), *info*: dict[str, Any], *loaded*: bool = False)

Bases: [Resource](#)

class `ironicclient.v1.shard.ShardManager`(*api*: [SessionClient](#))

Bases: [Manager](#)[[Shard](#)]

list(*os_ironic_api_version*: str | None = None, *global_request_id*: str | None = None) → list[[Shard](#)]

Retrieve a list of shards.

Parameters

- **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 [Shard](#)

ironicclient.v1.utils module

ironicclient.v1.volume_connector module

```
class ironicclient.v1.volume_connector.VolumeConnector(manager: ManagerProtocol,  
                                                         info: dict[str, Any], loaded:  
                                                         bool = False)
```

Bases: [Resource](#)

```
class ironicclient.v1.volume_connector.VolumeConnectorManager(api: SessionClient)
```

Bases: [CreateManager\[VolumeConnector\]](#)

```
delete(volume_connector_id: str, os_ironic_api_version: str | None = None,  
        global_request_id: str | None = None) → None
```

```
get(volume_connector_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None  
    = None, global_request_id: str | None = None) → Resource | None
```

```
list(node: str | None = None, limit: int | None = None, marker: str | None = None, sort_key:  
     str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None  
     = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None)  
     → list[VolumeConnector]
```

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: str, patch: list[dict[str, Any]], os_ironic_api_version: str | None = None, global_request_id: str | None = None) → [Resource](#)

ironicclient.v1.volume_target module

class ironicclient.v1.volume_target.**VolumeTarget**(manager: [ManagerProtocol](#), info: dict[str, Any], loaded: bool = False)

Bases: [Resource](#)

class ironicclient.v1.volume_target.**VolumeTargetManager**(api: [SessionClient](#))

Bases: [CreateManager](#)[[VolumeTarget](#)]

delete(volume_target_id: str, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → None

get(volume_target_id: str, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → [Resource](#) | None

list(node: str | None = None, limit: int | None = None, marker: str | None = None, sort_key: str | None = None, sort_dir: str | None = None, detail: bool = False, fields: list[str] | None = None, os_ironic_api_version: str | None = None, global_request_id: str | None = None) → list[[VolumeTarget](#)]

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: str, patch: list[dict[str, Any]], os_ironic_api_version: str | None = None, global_request_id: str | None = None*) → *Resource*

Module contents

Submodules

ironicclient.client module

ironicclient.client.Client(*version: str, endpoint_override: str | None = None, session: Session | None = None, *args: Any, **kwargs: Any*) → *Client*

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: str, auth_type: str | None = None, os_ironic_api_version: str | None = None, max_retries: int | None = None, retry_interval: int | None = None, session: Session | None = None, valid_interfaces: str | list[str] | None = None, interface: str | list[str] | None = None, service_type: str | None = None, region_name: str | None = None, additional_headers: dict[str, str] | None = None, global_request_id: str | None = None, **kwargs: Any*) → *Client*

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 *AmbiguousAuthSystem*

exception `ironicclient.exc.AmbiguousAuthSystem`

Bases: *ClientException*

Could not obtain token and endpoint using provided credentials.

exception `ironicclient.exc.InvalidAttribute`

Bases: *ClientException*

exception `ironicclient.exc.StateTransitionFailed`

Bases: *ClientException*

Failed to reach a requested provision state.

exception `ironicclient.exc.StateTransitionTimeout`

Bases: *ClientException*

Timed out while waiting for a requested provision state.

`ironicclient.exc.from_response(response: Response, message: str | None = None, traceback: str | None = None, method: str | None = None, url: str | None = None) → HttpError`

Return an *HttpError* instance based on response from http lib/requests.

ironicclient.shell module

class ironicclient.shell.App

Bases: App

build_option_parser(*description: str | None, version: str | None, argparse_kwargs: dict[str, object] | None = None*) → ArgumentParser

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: list[str]*) → None

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: CloudRegion, options: Namespace*)

Bases: object

property baremetal: [Client](#)

class ironicclient.shell.CommandManager(*namespace: str | None = None, convert_underscores: bool = True, *(Keyword-only parameters separator (PEP 3102)), ignored_modules: Iterable[str] | None = None*)

Bases: CommandManager

load_commands(*namespace: str*) → None

Load all the commands from an entrypoint

Parameters

namespace The namespace to load commands from.

Returns

None

ironicclient.shell.**main**(*argv: list[str] = ['-W', '-b', 'latex', 'doc/source', 'doc/build/pdf']*) → int

Module contents

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

C

- `ironicclient.client`, 208
- `ironicclient.common`, 119
- `ironicclient.common.apiclient`, 113
- `ironicclient.common.apiclient.base`, 93
- `ironicclient.common.apiclient.exceptions`, 93
- `ironicclient.common.base`, 113
- `ironicclient.common.filecache`, 114
- `ironicclient.common.http`, 114
- `ironicclient.common.i18n`, 115
- `ironicclient.common.utils`, 115

e

- `ironicclient.exc`, 209

i

- `ironicclient`, 210

O

- `ironicclient.osc`, 170
- `ironicclient.osc.command`, 169
- `ironicclient.osc.plugin`, 169
- `ironicclient.osc.v1`, 169
- `ironicclient.osc.v1.baremetal_allocation`, 119
- `ironicclient.osc.v1.baremetal_chassis`, 122
- `ironicclient.osc.v1.baremetal_conductor`, 124
- `ironicclient.osc.v1.baremetal_create`, 125
- `ironicclient.osc.v1.baremetal_deploy_template`, 126
- `ironicclient.osc.v1.baremetal_driver`, 129
- `ironicclient.osc.v1.baremetal_inspection_rule`, 132
- `ironicclient.osc.v1.baremetal_node`, 135
- `ironicclient.osc.v1.baremetal_port`, 154

- `ironicclient.osc.v1.baremetal_portgroup`, 156
- `ironicclient.osc.v1.baremetal_runbook`, 158
- `ironicclient.osc.v1.baremetal_shard`, 162
- `ironicclient.osc.v1.baremetal_volume_connector`, 162
- `ironicclient.osc.v1.baremetal_volume_target`, 165

S

- `ironicclient.shell`, 210

V

- `ironicclient.v1`, 208
- `ironicclient.v1.allocation`, 170
- `ironicclient.v1.chassis`, 172
- `ironicclient.v1.client`, 174
- `ironicclient.v1.conductor`, 175
- `ironicclient.v1.create_resources`, 176
- `ironicclient.v1.deploy_template`, 179
- `ironicclient.v1.driver`, 180
- `ironicclient.v1.events`, 182
- `ironicclient.v1.inspection_rule`, 182
- `ironicclient.v1.node`, 183
- `ironicclient.v1.port`, 197
- `ironicclient.v1.portgroup`, 198
- `ironicclient.v1.resource_fields`, 201
- `ironicclient.v1.runbook`, 203
- `ironicclient.v1.shard`, 205
- `ironicclient.v1.utils`, 205
- `ironicclient.v1.volume_connector`, 205
- `ironicclient.v1.volume_target`, 207

Symbols

- address
 - openstack-baremetal-port-group-create
 - command line option, 69
 - openstack-baremetal-port-group-list
 - command line option, 71
 - openstack-baremetal-port-group-set
 - command line option, 72
 - openstack-baremetal-port-group-show
 - command line option, 73
 - openstack-baremetal-port-group-unset
 - command line option, 73
 - openstack-baremetal-port-list
 - command line option, 75
 - openstack-baremetal-port-set
 - command line option, 76
 - openstack-baremetal-port-show
 - command line option, 77
- all
 - openstack-baremetal-node-remove-trait
 - command line option, 51
 - openstack-baremetal-node-trait-remove
 - command line option, 59
- arg
 - openstack-baremetal-driver-passthru-call
 - command line option, 25
 - openstack-baremetal-node-passthru-call
 - command line option, 48
- associated
 - openstack-baremetal-node-list
 - command line option, 46
- automated-clean
 - openstack-baremetal-node-create
 - command line option, 39
 - openstack-baremetal-node-set
 - command line option, 56
 - openstack-baremetal-node-unset
 - command line option, 63
- bios-interface
 - openstack-baremetal-node-create
 - command line option, 38
- openstack-baremetal-node-set
 - command line option, 54
- openstack-baremetal-node-unset
 - command line option, 62
- boot-index
 - openstack-baremetal-volume-target-create
 - command line option, 84
 - openstack-baremetal-volume-target-set
 - command line option, 86
- boot-interface
 - openstack-baremetal-node-create
 - command line option, 38
 - openstack-baremetal-node-set
 - command line option, 54
 - openstack-baremetal-node-unset
 - command line option, 62
- candidate-node
 - openstack-baremetal-allocation-create
 - command line option, 8
- category
 - openstack-baremetal-port-create
 - command line option, 68
 - openstack-baremetal-port-group-create
 - command line option, 69
 - openstack-baremetal-port-group-set
 - command line option, 72
 - openstack-baremetal-port-group-unset
 - command line option, 74
 - openstack-baremetal-port-set
 - command line option, 76
 - openstack-baremetal-port-unset
 - command line option, 78
- chassis
 - openstack-baremetal-node-list
 - command line option, 46
- chassis-uuid
 - openstack-baremetal-node-create
 - command line option, 37
 - openstack-baremetal-node-set
 - command line option, 54
 - openstack-baremetal-node-unset
 - command line option, 62

command line option, 62

--clean-steps

- openstack-baremetal-node-clean
 - command line option, 34

--column

- openstack-baremetal-allocation-create
 - command line option, 8
- openstack-baremetal-allocation-list
 - command line option, 9
- openstack-baremetal-allocation-show
 - command line option, 11
- openstack-baremetal-chassis-create
 - command line option, 12
- openstack-baremetal-chassis-list
 - command line option, 14
- openstack-baremetal-chassis-show
 - command line option, 15
- openstack-baremetal-conductor-list
 - command line option, 17
- openstack-baremetal-conductor-show
 - command line option, 18
- openstack-baremetal-deploy-template-create
 - command line option, 19
- openstack-baremetal-deploy-template-list
 - command line option, 20
- openstack-baremetal-deploy-template-show
 - command line option, 22
- openstack-baremetal-driver-list
 - command line option, 23
- openstack-baremetal-driver-passthru-call
 - command line option, 25
- openstack-baremetal-driver-passthru-list
 - command line option, 26
- openstack-baremetal-driver-property-list
 - command line option, 27
- openstack-baremetal-driver-raid-property-list
 - command line option, 28
- openstack-baremetal-driver-show
 - command line option, 29
- openstack-baremetal-node-bios-setting-list
 - command line option, 30
- openstack-baremetal-node-bios-setting-show
 - command line option, 31
- openstack-baremetal-node-boot-device-show
 - command line option, 33
- openstack-baremetal-node-children-list
 - command line option, 34
- openstack-baremetal-node-console-show
 - command line option, 35
- openstack-baremetal-node-create
 - command line option, 37
- openstack-baremetal-node-firmware-list
 - command line option, 40
- openstack-baremetal-node-history-get
 - command line option, 41
- openstack-baremetal-node-history-list
 - command line option, 42
- openstack-baremetal-node-list
 - command line option, 45
- openstack-baremetal-node-passthru-list
 - command line option, 48
- openstack-baremetal-node-show
 - command line option, 57
- openstack-baremetal-node-trait-list
 - command line option, 58
- openstack-baremetal-node-validate
 - command line option, 64
- openstack-baremetal-node-vif-list
 - command line option, 65
- openstack-baremetal-port-create
 - command line option, 67
- openstack-baremetal-port-group-create
 - command line option, 68
- openstack-baremetal-port-group-list
 - command line option, 70
- openstack-baremetal-port-group-show
 - command line option, 73
- openstack-baremetal-port-list
 - command line option, 74
- openstack-baremetal-port-show
 - command line option, 77
- openstack-baremetal-volume-connector-create
 - command line option, 79
- openstack-baremetal-volume-connector-list
 - command line option, 80
- openstack-baremetal-volume-connector-show
 - command line option, 82
- openstack-baremetal-volume-target-create
 - command line option, 83
- openstack-baremetal-volume-target-list
 - command line option, 85
- openstack-baremetal-volume-target-show
 - command line option, 87

--conductor

- openstack-baremetal-node-list
 - command line option, 46
- openstack-baremetal-node-create
 - command line option, 39
- openstack-baremetal-node-list
 - command line option, 46
- openstack-baremetal-node-set

- command line option, 56
- openstack-baremetal-node-unset
 - command line option, 62
- config-drive
 - openstack-baremetal-node-deploy
 - command line option, 40
 - openstack-baremetal-node-rebuild
 - command line option, 51
- connector-id
 - openstack-baremetal-volume-connector-create
 - command line option, 79
 - openstack-baremetal-volume-connector-set
 - command line option, 81
- console-interface
 - openstack-baremetal-node-create
 - command line option, 38
 - openstack-baremetal-node-set
 - command line option, 54
 - openstack-baremetal-node-unset
 - command line option, 62
- deploy-interface
 - openstack-baremetal-node-create
 - command line option, 38
 - openstack-baremetal-node-set
 - command line option, 54
 - openstack-baremetal-node-unset
 - command line option, 62
- deploy-steps
 - openstack-baremetal-node-deploy
 - command line option, 40
 - openstack-baremetal-node-rebuild
 - command line option, 51
- description
 - openstack-baremetal-chassis-create
 - command line option, 13
 - openstack-baremetal-chassis-set
 - command line option, 15
 - openstack-baremetal-chassis-unset
 - command line option, 16
 - openstack-baremetal-node-create
 - command line option, 39
 - openstack-baremetal-node-set
 - command line option, 57
 - openstack-baremetal-node-unset
 - command line option, 63
 - openstack-baremetal-port-create
 - command line option, 67
 - openstack-baremetal-port-set
 - command line option, 76
 - openstack-baremetal-port-unset
 - command line option, 78
- description-contains
 - openstack-baremetal-node-list
 - command line option, 46
- disable-power-off
 - openstack-baremetal-node-create
 - command line option, 39
 - openstack-baremetal-node-set
 - command line option, 57
- disable-ramdisk
 - openstack-baremetal-node-clean
 - command line option, 35
 - openstack-baremetal-node-service
 - command line option, 53
- driver
 - openstack-baremetal-node-create
 - command line option, 37
 - openstack-baremetal-node-list
 - command line option, 46
 - openstack-baremetal-node-set
 - command line option, 54
- driver-info
 - openstack-baremetal-node-create
 - command line option, 37
 - openstack-baremetal-node-set
 - command line option, 56
 - openstack-baremetal-node-unset
 - command line option, 62
- enable-power-off
 - openstack-baremetal-node-set
 - command line option, 57
- extra
 - openstack-baremetal-allocation-create
 - command line option, 8
 - openstack-baremetal-allocation-set
 - command line option, 11
 - openstack-baremetal-allocation-unset
 - command line option, 12
 - openstack-baremetal-chassis-create
 - command line option, 13
 - openstack-baremetal-chassis-set
 - command line option, 15
 - openstack-baremetal-chassis-unset
 - command line option, 16
 - openstack-baremetal-deploy-template-create
 - command line option, 20
 - openstack-baremetal-deploy-template-set
 - command line option, 22
 - openstack-baremetal-deploy-template-unset
 - command line option, 23
 - openstack-baremetal-node-create
 - command line option, 37

openstack-baremetal-node-set
 command line option, [56](#)
openstack-baremetal-node-unset
 command line option, [61](#)
openstack-baremetal-port-create
 command line option, [67](#)
openstack-baremetal-port-group-create
 command line option, [69](#)
openstack-baremetal-port-group-set
 command line option, [72](#)
openstack-baremetal-port-group-unset
 command line option, [73](#)
openstack-baremetal-port-set
 command line option, [76](#)
openstack-baremetal-port-unset
 command line option, [78](#)
openstack-baremetal-volume-connector-create
 command line option, [79](#)
openstack-baremetal-volume-connector-set
 command line option, [82](#)
openstack-baremetal-volume-connector-unset
 command line option, [83](#)
openstack-baremetal-volume-target-create
 command line option, [84](#)
openstack-baremetal-volume-target-set
 command line option, [86](#)
openstack-baremetal-volume-target-unset
 command line option, [88](#)
--fault
 openstack-baremetal-node-list
 command line option, [46](#)
--fields
 openstack-baremetal-allocation-list
 command line option, [10](#)
 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-conductor-list
 command line option, [17](#)
 openstack-baremetal-conductor-show
 command line option, [18](#)
 openstack-baremetal-deploy-template-list
 command line option, [21](#)
 openstack-baremetal-deploy-template-show
 command line option, [23](#)
 openstack-baremetal-driver-list
 command line option, [24](#)
 openstack-baremetal-driver-show
 command line option, [29](#)
openstack-baremetal-node-bios-setting-list
 command line option, [31](#)
openstack-baremetal-node-list
 command line option, [46](#)
openstack-baremetal-node-show
 command line option, [58](#)
openstack-baremetal-port-group-list
 command line option, [71](#)
openstack-baremetal-port-group-show
 command line option, [73](#)
openstack-baremetal-port-list
 command line option, [75](#)
openstack-baremetal-port-show
 command line option, [77](#)
openstack-baremetal-volume-connector-list
 command line option, [81](#)
openstack-baremetal-volume-connector-show
 command line option, [82](#)
openstack-baremetal-volume-target-list
 command line option, [86](#)
openstack-baremetal-volume-target-show
 command line option, [87](#)
--file
openstack-baremetal-node-inventory-save
 command line option, [44](#)
firmware-interface
openstack-baremetal-node-create
 command line option, [39](#)
openstack-baremetal-node-set
 command line option, [55](#)
openstack-baremetal-node-unset
 command line option, [62](#)
--fit-width
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-conductor-list
 command line option, [17](#)
openstack-baremetal-conductor-show
 command line option, [18](#)
openstack-baremetal-deploy-template-create

- command line option, [19](#)
- openstack-baremetal-deploy-template-list
 - command line option, [21](#)
- openstack-baremetal-deploy-template-show
 - command line option, [22](#)
- openstack-baremetal-driver-list
 - command line option, [24](#)
- openstack-baremetal-driver-passthru-call
 - command line option, [25](#)
- openstack-baremetal-driver-passthru-list
 - command line option, [26](#)
- openstack-baremetal-driver-property-list
 - command line option, [27](#)
- openstack-baremetal-driver-raid-property-list
 - command line option, [28](#)
- openstack-baremetal-driver-show
 - command line option, [29](#)
- openstack-baremetal-node-bios-setting-list
 - command line option, [30](#)
- openstack-baremetal-node-bios-setting-show
 - command line option, [32](#)
- openstack-baremetal-node-boot-device-show
 - command line option, [33](#)
- openstack-baremetal-node-children-list
 - command line option, [34](#)
- openstack-baremetal-node-console-show
 - command line option, [36](#)
- openstack-baremetal-node-create
 - command line option, [37](#)
- openstack-baremetal-node-firmware-list
 - command line option, [41](#)
- openstack-baremetal-node-history-get
 - command line option, [42](#)
- openstack-baremetal-node-history-list
 - command line option, [43](#)
- openstack-baremetal-node-list
 - command line option, [45](#)
- openstack-baremetal-node-passthru-list
 - command line option, [49](#)
- openstack-baremetal-node-show
 - command line option, [57](#)
- openstack-baremetal-node-trait-list
 - command line option, [59](#)
- openstack-baremetal-node-validate
 - command line option, [64](#)
- openstack-baremetal-node-vif-list
 - command line option, [66](#)
- openstack-baremetal-port-create
 - command line option, [67](#)
- openstack-baremetal-port-group-create
 - command line option, [69](#)
- openstack-baremetal-port-group-list
 - command line option, [70](#)
- openstack-baremetal-port-group-show
 - command line option, [73](#)
- openstack-baremetal-port-list
 - command line option, [74](#)
- openstack-baremetal-port-show
 - command line option, [77](#)
- openstack-baremetal-volume-connector-create
 - command line option, [79](#)
- openstack-baremetal-volume-connector-list
 - command line option, [80](#)
- openstack-baremetal-volume-connector-show
 - command line option, [82](#)
- openstack-baremetal-volume-target-create
 - command line option, [84](#)
- openstack-baremetal-volume-target-list
 - command line option, [85](#)
- openstack-baremetal-volume-target-show
 - command line option, [87](#)
- format
- openstack-baremetal-allocation-create
 - command line option, [8](#)
- openstack-baremetal-allocation-list
 - command line option, [9](#)
- openstack-baremetal-allocation-show
 - command line option, [11](#)
- openstack-baremetal-chassis-create
 - command line option, [12](#)
- openstack-baremetal-chassis-list
 - command line option, [14](#)
- openstack-baremetal-chassis-show
 - command line option, [15](#)
- openstack-baremetal-conductor-list
 - command line option, [17](#)
- openstack-baremetal-conductor-show
 - command line option, [18](#)
- openstack-baremetal-deploy-template-create
 - command line option, [19](#)
- openstack-baremetal-deploy-template-list
 - command line option, [20](#)
- openstack-baremetal-deploy-template-show
 - command line option, [22](#)
- openstack-baremetal-driver-list
 - command line option, [23](#)
- openstack-baremetal-driver-passthru-call
 - command line option, [24](#)
- openstack-baremetal-driver-passthru-list
 - command line option, [25](#)
- openstack-baremetal-driver-property-list
 - command line option, [26](#)

openstack-baremetal-driver-raid-property-list [command line option, 82](#)
 [command line option, 27](#) openstack-baremetal-volume-target-create
openstack-baremetal-driver-show [command line option, 83](#)
 [command line option, 28](#) openstack-baremetal-volume-target-list
openstack-baremetal-node-bios-setting-list [command line option, 85](#)
 [command line option, 30](#) openstack-baremetal-volume-target-show
openstack-baremetal-node-bios-setting-show [command line option, 87](#)
 [command line option, 31](#) --http-method
openstack-baremetal-node-boot-device-show openstack-baremetal-driver-passthru-call
 [command line option, 32](#) [command line option, 25](#)
openstack-baremetal-node-children-list openstack-baremetal-node-passthru-call
 [command line option, 34](#) [command line option, 48](#)
openstack-baremetal-node-console-show --include-children
 [command line option, 35](#) openstack-baremetal-node-list
openstack-baremetal-node-create [command line option, 47](#)
 [command line option, 37](#) --inspect-interface
openstack-baremetal-node-firmware-list openstack-baremetal-node-create
 [command line option, 40](#) [command line option, 38](#)
openstack-baremetal-node-history-get openstack-baremetal-node-set
 [command line option, 41](#) [command line option, 55](#)
openstack-baremetal-node-history-list openstack-baremetal-node-unset
 [command line option, 42](#) [command line option, 62](#)
openstack-baremetal-node-list --instance
 [command line option, 45](#) openstack-baremetal-node-show
openstack-baremetal-node-passthru-list [command line option, 58](#)
 [command line option, 48](#) --instance-info
openstack-baremetal-node-show openstack-baremetal-node-set
 [command line option, 57](#) [command line option, 56](#)
openstack-baremetal-node-trait-list openstack-baremetal-node-unset
 [command line option, 58](#) [command line option, 62](#)
openstack-baremetal-node-validate --instance-name
 [command line option, 63](#) openstack-baremetal-node-create
openstack-baremetal-node-vif-list [command line option, 38](#)
 [command line option, 65](#) openstack-baremetal-node-list
openstack-baremetal-port-create [command line option, 46](#)
 [command line option, 66](#) openstack-baremetal-node-set
openstack-baremetal-port-group-create [command line option, 54](#)
 [command line option, 68](#) openstack-baremetal-node-unset
openstack-baremetal-port-group-list [command line option, 61](#)
 [command line option, 70](#) --instance-uuid
openstack-baremetal-port-group-show openstack-baremetal-node-set
 [command line option, 72](#) [command line option, 54](#)
openstack-baremetal-port-list openstack-baremetal-node-unset
 [command line option, 74](#) [command line option, 61](#)
openstack-baremetal-port-show --is-smartnic
 [command line option, 77](#) openstack-baremetal-port-create
openstack-baremetal-volume-connector-create [command line option, 67](#)
 [command line option, 79](#) openstack-baremetal-port-set
openstack-baremetal-volume-connector-list [command line option, 76](#)
 [command line option, 80](#) openstack-baremetal-port-unset
openstack-baremetal-volume-connector-show [command line option, 78](#)

- lessee
 - openstack-baremetal-node-create
 - command line option, 39
 - openstack-baremetal-node-list
 - command line option, 46
 - openstack-baremetal-node-set
 - command line option, 56
 - openstack-baremetal-node-unset
 - command line option, 63
- limit
 - openstack-baremetal-allocation-list
 - command line option, 10
 - openstack-baremetal-chassis-list
 - command line option, 14
 - openstack-baremetal-conductor-list
 - command line option, 17
 - openstack-baremetal-deploy-template-list
 - command line option, 21
 - openstack-baremetal-node-list
 - command line option, 45
 - openstack-baremetal-port-group-list
 - command line option, 71
 - openstack-baremetal-port-list
 - command line option, 75
 - openstack-baremetal-volume-connector-list
 - command line option, 81
 - openstack-baremetal-volume-target-list
 - command line option, 85
- local-link-connection
 - openstack-baremetal-port-create
 - command line option, 67
 - openstack-baremetal-port-set
 - command line option, 76
 - openstack-baremetal-port-unset
 - command line option, 78
- long
 - openstack-baremetal-allocation-list
 - command line option, 10
 - openstack-baremetal-chassis-list
 - command line option, 14
 - openstack-baremetal-conductor-list
 - command line option, 17
 - openstack-baremetal-deploy-template-list
 - command line option, 21
 - openstack-baremetal-driver-list
 - command line option, 24
 - openstack-baremetal-node-bios-setting-list
 - command line option, 31
 - openstack-baremetal-node-history-list
 - command line option, 43
 - openstack-baremetal-node-list
 - command line option, 46
 - openstack-baremetal-port-group-list
 - command line option, 71
 - openstack-baremetal-port-list
 - command line option, 75
 - openstack-baremetal-volume-connector-list
 - command line option, 81
 - openstack-baremetal-volume-target-list
 - command line option, 86
- maintenance
 - openstack-baremetal-node-list
 - command line option, 45
- management-interface
 - openstack-baremetal-node-create
 - command line option, 38
 - openstack-baremetal-node-set
 - command line option, 55
 - openstack-baremetal-node-unset
 - command line option, 62
- marker
 - openstack-baremetal-allocation-list
 - command line option, 10
 - openstack-baremetal-chassis-list
 - command line option, 14
 - openstack-baremetal-conductor-list
 - command line option, 17
 - openstack-baremetal-deploy-template-list
 - command line option, 21
 - openstack-baremetal-node-list
 - command line option, 45
 - openstack-baremetal-port-group-list
 - command line option, 71
 - openstack-baremetal-port-list
 - command line option, 75
 - openstack-baremetal-volume-connector-list
 - command line option, 81
 - openstack-baremetal-volume-target-list
 - command line option, 86
- max-width
 - 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

<code>openstack-baremetal-conductor-list</code>	command line option, 65
command line option, 17	<code>openstack-baremetal-port-create</code>
<code>openstack-baremetal-conductor-show</code>	command line option, 67
command line option, 18	<code>openstack-baremetal-port-group-create</code>
<code>openstack-baremetal-deploy-template-create</code>	command line option, 69
command line option, 19	<code>openstack-baremetal-port-group-list</code>
<code>openstack-baremetal-deploy-template-list</code>	command line option, 70
command line option, 21	<code>openstack-baremetal-port-group-show</code>
<code>openstack-baremetal-deploy-template-show</code>	command line option, 73
command line option, 22	<code>openstack-baremetal-port-list</code>
<code>openstack-baremetal-driver-list</code>	command line option, 74
command line option, 24	<code>openstack-baremetal-port-show</code>
<code>openstack-baremetal-driver-passthru-call</code>	command line option, 77
command line option, 25	<code>openstack-baremetal-volume-connector-create</code>
<code>openstack-baremetal-driver-passthru-list</code>	command line option, 79
command line option, 26	<code>openstack-baremetal-volume-connector-list</code>
<code>openstack-baremetal-driver-property-list</code>	command line option, 80
command line option, 27	<code>openstack-baremetal-volume-connector-show</code>
<code>openstack-baremetal-driver-raid-property-list</code>	command line option, 82
command line option, 28	<code>openstack-baremetal-volume-target-create</code>
<code>openstack-baremetal-driver-show</code>	command line option, 83
command line option, 29	<code>openstack-baremetal-volume-target-list</code>
<code>openstack-baremetal-node-bios-setting-list</code>	command line option, 85
command line option, 30	<code>openstack-baremetal-volume-target-show</code>
<code>openstack-baremetal-node-bios-setting-show</code>	command line option, 87
command line option, 31	<code>--mode</code>
<code>openstack-baremetal-node-boot-device-show</code>	<code>openstack-baremetal-port-group-create</code>
command line option, 33	command line option, 69
<code>openstack-baremetal-node-children-list</code>	<code>openstack-baremetal-port-group-set</code>
command line option, 34	command line option, 72
<code>openstack-baremetal-node-console-show --name</code>	<code>openstack-baremetal-allocation-create</code>
command line option, 36	command line option, 8
<code>openstack-baremetal-node-create</code>	<code>openstack-baremetal-allocation-set</code>
command line option, 37	command line option, 11
<code>openstack-baremetal-node-firmware-list</code>	<code>openstack-baremetal-allocation-unset</code>
command line option, 41	command line option, 12
<code>openstack-baremetal-node-history-get</code>	<code>openstack-baremetal-deploy-template-set</code>
command line option, 42	command line option, 22
<code>openstack-baremetal-node-history-list</code>	<code>openstack-baremetal-node-create</code>
command line option, 43	command line option, 38
<code>openstack-baremetal-node-list</code>	<code>openstack-baremetal-node-set</code>
command line option, 45	command line option, 54
<code>openstack-baremetal-node-passthru-list</code>	<code>openstack-baremetal-node-unset</code>
command line option, 49	command line option, 61
<code>openstack-baremetal-node-show</code>	<code>openstack-baremetal-port-create</code>
command line option, 57	command line option, 67
<code>openstack-baremetal-node-trait-list</code>	<code>openstack-baremetal-port-group-create</code>
command line option, 59	command line option, 69
<code>openstack-baremetal-node-validate</code>	<code>openstack-baremetal-port-group-set</code>
command line option, 64	command line option, 72
<code>openstack-baremetal-node-vif-list</code>	

- openstack-baremetal-port-group-unset
 - command line option, [73](#)
- openstack-baremetal-port-set
 - command line option, [76](#)
- openstack-baremetal-port-unset
 - command line option, [78](#)
- network-data
 - openstack-baremetal-node-create
 - command line option, [38](#)
 - openstack-baremetal-node-set
 - command line option, [55](#)
 - openstack-baremetal-node-unset
 - command line option, [62](#)
- network-interface
 - openstack-baremetal-node-create
 - command line option, [38](#)
 - openstack-baremetal-node-set
 - command line option, [55](#)
 - openstack-baremetal-node-unset
 - command line option, [62](#)
- no-automated-clean
 - openstack-baremetal-node-create
 - command line option, [39](#)
 - openstack-baremetal-node-set
 - command line option, [56](#)
- no-maintenance
 - openstack-baremetal-node-list
 - command line option, [45](#)
- no-retired
 - openstack-baremetal-node-list
 - command line option, [46](#)
- node
 - openstack-baremetal-allocation-create
 - command line option, [9](#)
 - openstack-baremetal-allocation-list
 - command line option, [10](#)
 - openstack-baremetal-port-create
 - command line option, [67](#)
 - openstack-baremetal-port-group-create
 - command line option, [69](#)
 - openstack-baremetal-port-group-list
 - command line option, [71](#)
 - openstack-baremetal-port-group-set
 - command line option, [71](#)
 - openstack-baremetal-port-list
 - command line option, [75](#)
 - openstack-baremetal-port-set
 - command line option, [76](#)
 - openstack-baremetal-volume-connector-create
 - command line option, [79](#)
 - openstack-baremetal-volume-connector-list
 - command line option, [81](#)
 - openstack-baremetal-volume-connector-set
 - command line option, [81](#)
 - openstack-baremetal-volume-target-create
 - command line option, [84](#)
 - openstack-baremetal-volume-target-list
 - command line option, [85](#)
 - openstack-baremetal-volume-target-set
 - command line option, [86](#)
- noindent
 - openstack-baremetal-allocation-create
 - command line option, [8](#)
 - openstack-baremetal-allocation-list
 - command line option, [9](#)
 - openstack-baremetal-allocation-show
 - command line option, [11](#)
 - openstack-baremetal-chassis-create
 - command line option, [12](#)
 - openstack-baremetal-chassis-list
 - command line option, [14](#)
 - openstack-baremetal-chassis-show
 - command line option, [15](#)
 - openstack-baremetal-conductor-list
 - command line option, [17](#)
 - openstack-baremetal-conductor-show
 - command line option, [18](#)
 - openstack-baremetal-deploy-template-create
 - command line option, [19](#)
 - openstack-baremetal-deploy-template-list
 - command line option, [21](#)
 - openstack-baremetal-deploy-template-show
 - command line option, [22](#)
 - openstack-baremetal-driver-list
 - command line option, [23](#)
 - openstack-baremetal-driver-passthru-call
 - command line option, [25](#)
 - openstack-baremetal-driver-passthru-list
 - command line option, [26](#)
 - openstack-baremetal-driver-property-list
 - command line option, [27](#)
 - openstack-baremetal-driver-raid-property-list
 - command line option, [28](#)
 - openstack-baremetal-driver-show
 - command line option, [29](#)
 - openstack-baremetal-node-bios-setting-list
 - command line option, [30](#)
 - openstack-baremetal-node-bios-setting-show
 - command line option, [31](#)
 - openstack-baremetal-node-boot-device-show
 - command line option, [33](#)
 - openstack-baremetal-node-children-list
 - command line option, [33](#)

command line option, 34	command line option, 10
openstack-baremetal-node-console-show	openstack-baremetal-node-create
command line option, 35	command line option, 39
openstack-baremetal-node-create	openstack-baremetal-node-list
command line option, 37	command line option, 46
openstack-baremetal-node-firmware-list	openstack-baremetal-node-set
command line option, 41	command line option, 56
openstack-baremetal-node-history-get	openstack-baremetal-node-unset
command line option, 42	command line option, 63
openstack-baremetal-node-history-list--parent-node	
command line option, 42	openstack-baremetal-node-create
openstack-baremetal-node-list	command line option, 39
command line option, 45	openstack-baremetal-node-list
openstack-baremetal-node-passthru-list	command line option, 47
command line option, 48	openstack-baremetal-node-set
openstack-baremetal-node-show	command line option, 57
command line option, 57	openstack-baremetal-node-unset
openstack-baremetal-node-trait-list	command line option, 63
command line option, 59	--persistent
openstack-baremetal-node-validate	openstack-baremetal-node-boot-device-set
command line option, 64	command line option, 32
openstack-baremetal-node-vif-list	--physical-network
command line option, 65	openstack-baremetal-port-create
openstack-baremetal-port-create	command line option, 67
command line option, 67	openstack-baremetal-port-group-create
openstack-baremetal-port-group-create	command line option, 69
command line option, 68	openstack-baremetal-port-group-set
openstack-baremetal-port-group-list	command line option, 72
command line option, 70	openstack-baremetal-port-group-unset
openstack-baremetal-port-group-show	command line option, 74
command line option, 73	openstack-baremetal-port-set
openstack-baremetal-port-list	command line option, 76
command line option, 74	openstack-baremetal-port-unset
openstack-baremetal-port-show	command line option, 78
command line option, 77	--port-group
openstack-baremetal-volume-connector-create	openstack-baremetal-port-create
command line option, 79	command line option, 67
openstack-baremetal-volume-connector-list	openstack-baremetal-port-list
command line option, 80	command line option, 75
openstack-baremetal-volume-connector-show	openstack-baremetal-port-set
command line option, 82	command line option, 76
openstack-baremetal-volume-target-create	openstack-baremetal-port-unset
command line option, 83	command line option, 78
openstack-baremetal-volume-target-list--port-uuid	
command line option, 85	openstack-baremetal-node-vif-attach
openstack-baremetal-volume-target-show	command line option, 64
command line option, 87	--portgroup-uuid
--owner	openstack-baremetal-node-vif-attach
openstack-baremetal-allocation-create	command line option, 64
command line option, 8	--power-interface
openstack-baremetal-allocation-list	openstack-baremetal-node-create

command line option, 38
 openstack-baremetal-node-set
 command line option, 55
 openstack-baremetal-node-unset
 command line option, 62
 --power-timeout
 openstack-baremetal-node-power-off
 command line option, 49
 openstack-baremetal-node-power-on
 command line option, 50
 openstack-baremetal-node-reboot
 command line option, 50
 --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-conductor-show
 command line option, 18
 openstack-baremetal-deploy-template-create
 command line option, 19
 openstack-baremetal-deploy-template-show
 command line option, 22
 openstack-baremetal-driver-passthru-call
 command line option, 25
 openstack-baremetal-driver-show
 command line option, 29
 openstack-baremetal-node-bios-setting-show
 command line option, 31
 openstack-baremetal-node-boot-device-show
 command line option, 33
 openstack-baremetal-node-children-list
 command line option, 34
 openstack-baremetal-node-console-show
 command line option, 36
 openstack-baremetal-node-create
 command line option, 37
 openstack-baremetal-node-history-get
 command line option, 42
 openstack-baremetal-node-show
 command line option, 57
 openstack-baremetal-port-create
 command line option, 67
 openstack-baremetal-port-group-create
 command line option, 68
 openstack-baremetal-port-group-show
 command line option, 73
 openstack-baremetal-port-show
 command line option, 77
 openstack-baremetal-volume-connector-create
 command line option, 79
 openstack-baremetal-volume-connector-show
 command line option, 82
 openstack-baremetal-volume-target-create
 command line option, 83
 openstack-baremetal-volume-target-show
 command line option, 87
 --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, 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-conductor-list
 command line option, 17
 openstack-baremetal-conductor-show
 command line option, 18
 openstack-baremetal-deploy-template-create
 command line option, 19
 openstack-baremetal-deploy-template-list
 command line option, 21
 openstack-baremetal-deploy-template-show
 command line option, 22
 openstack-baremetal-driver-list
 command line option, 24
 openstack-baremetal-driver-passthru-call
 command line option, 25
 openstack-baremetal-driver-passthru-list
 command line option, 26
 openstack-baremetal-driver-property-list
 command line option, 27
 openstack-baremetal-driver-raid-property-list
 command line option, 28
 openstack-baremetal-driver-show
 command line option, 29
 openstack-baremetal-node-bios-setting-list
 command line option, 30
 openstack-baremetal-node-bios-setting-show
 command line option, 32
 openstack-baremetal-node-boot-device-show
 command line option, 33

openstack-baremetal-node-children-list	openstack-baremetal-node-set
command line option, 34	command line option, 56
openstack-baremetal-node-console-show	openstack-baremetal-node-unset
command line option, 36	command line option, 61
openstack-baremetal-node-create	openstack-baremetal-port-group-create
command line option, 37	command line option, 69
openstack-baremetal-node-firmware-list	openstack-baremetal-port-group-set
command line option, 41	command line option, 72
openstack-baremetal-node-history-get	openstack-baremetal-port-group-unset
command line option, 42	command line option, 74
openstack-baremetal-node-history-list	openstack-baremetal-volume-target-create
command line option, 43	command line option, 84
openstack-baremetal-node-list	openstack-baremetal-volume-target-set
command line option, 45	command line option, 86
openstack-baremetal-node-passthru-list	openstack-baremetal-volume-target-unset
command line option, 49	command line option, 88
openstack-baremetal-node-show	--protected
command line option, 58	openstack-baremetal-node-set
openstack-baremetal-node-trait-list	command line option, 56
command line option, 59	openstack-baremetal-node-unset
openstack-baremetal-node-validate	command line option, 63
command line option, 64	--protected-reason
openstack-baremetal-node-vif-list	openstack-baremetal-node-set
command line option, 66	command line option, 56
openstack-baremetal-port-create	openstack-baremetal-node-unset
command line option, 67	command line option, 63
openstack-baremetal-port-group-create	--provision-state
command line option, 69	openstack-baremetal-node-list
openstack-baremetal-port-group-list	command line option, 46
command line option, 70	--pxe-disabled
openstack-baremetal-port-group-show	openstack-baremetal-port-set
command line option, 73	command line option, 76
openstack-baremetal-port-list	--pxe-enabled
command line option, 75	openstack-baremetal-port-create
openstack-baremetal-port-show	command line option, 67
command line option, 77	openstack-baremetal-port-set
openstack-baremetal-volume-connector-create	command line option, 76
command line option, 79	--quote
openstack-baremetal-volume-connector-list	openstack-baremetal-allocation-list
command line option, 80	command line option, 9
openstack-baremetal-volume-connector-show	openstack-baremetal-chassis-list
command line option, 82	command line option, 14
openstack-baremetal-volume-target-create	openstack-baremetal-conductor-list
command line option, 84	command line option, 17
openstack-baremetal-volume-target-list	openstack-baremetal-deploy-template-list
command line option, 85	command line option, 20
openstack-baremetal-volume-target-show	openstack-baremetal-driver-list
command line option, 87	command line option, 23
--property	openstack-baremetal-driver-passthru-list
openstack-baremetal-node-create	command line option, 26
command line option, 37	openstack-baremetal-driver-property-list

[command line option, 27](#)
[openstack-baremetal-driver-raid-property-list](#)
[command line option, 28](#)
[openstack-baremetal-node-bios-setting-list](#)
[command line option, 30](#)
[openstack-baremetal-node-firmware-list](#)
[command line option, 41](#)
[openstack-baremetal-node-history-list](#)
[command line option, 42](#)
[openstack-baremetal-node-list](#)
[command line option, 45](#)
[openstack-baremetal-node-passthru-list](#)
[command line option, 48](#)
[openstack-baremetal-node-trait-list](#)
[command line option, 58](#)
[openstack-baremetal-node-validate](#)
[command line option, 64](#)
[openstack-baremetal-node-vif-list](#)
[command line option, 65](#)
[openstack-baremetal-port-group-list](#)
[command line option, 70](#)
[openstack-baremetal-port-list](#)
[command line option, 74](#)
[openstack-baremetal-volume-connector-list](#)
[command line option, 80](#)
[openstack-baremetal-volume-target-list](#)
[command line option, 85](#)
[--raid-interface](#)
[openstack-baremetal-node-create](#)
[command line option, 38](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[openstack-baremetal-node-unset](#)
[command line option, 62](#)
[--reason](#)
[openstack-baremetal-node-maintenance-set](#)
[command line option, 47](#)
[--rescue-interface](#)
[openstack-baremetal-node-create](#)
[command line option, 38](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[openstack-baremetal-node-unset](#)
[command line option, 62](#)
[--rescue-password](#)
[openstack-baremetal-node-rescue](#)
[command line option, 52](#)
[--reset-bios-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 54](#)
[--reset-boot-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 54](#)
[--reset-console-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 54](#)
[--reset-deploy-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-firmware-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-interfaces](#)
[openstack-baremetal-node-set](#)
[command line option, 56](#)
[--reset-management-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-network-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-power-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-raid-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-rescue-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-storage-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 55](#)
[--reset-vendor-interface](#)
[openstack-baremetal-node-set](#)
[command line option, 56](#)
[--resource-class](#)
[openstack-baremetal-allocation-create](#)
[command line option, 8](#)
[openstack-baremetal-allocation-list](#)
[command line option, 10](#)
[openstack-baremetal-node-create](#)
[command line option, 39](#)
[openstack-baremetal-node-list](#)
[command line option, 46](#)
[openstack-baremetal-node-set](#)
[command line option, 56](#)
[openstack-baremetal-node-unset](#)
[command line option, 61](#)

--retired
 openstack-baremetal-node-list
 command line option, [45](#)
 openstack-baremetal-node-set
 command line option, [56](#)
 openstack-baremetal-node-unset
 command line option, [63](#)
--retired-reason
 openstack-baremetal-node-set
 command line option, [56](#)
 openstack-baremetal-node-unset
 command line option, [63](#)
--runbook
 openstack-baremetal-node-clean
 command line option, [34](#)
 openstack-baremetal-node-service
 command line option, [53](#)
--service-steps
 openstack-baremetal-node-service
 command line option, [53](#)
--shard
 openstack-baremetal-node-create
 command line option, [39](#)
 openstack-baremetal-node-set
 command line option, [57](#)
 openstack-baremetal-node-unset
 command line option, [63](#)
--sharded
 openstack-baremetal-node-list
 command line option, [46](#)
--shards
 openstack-baremetal-node-list
 command line option, [46](#)
 openstack-baremetal-port-group-list
 command line option, [71](#)
--soft
 openstack-baremetal-node-power-off
 command line option, [49](#)
 openstack-baremetal-node-reboot
 command line option, [50](#)
--sort
 openstack-baremetal-allocation-list
 command line option, [10](#)
 openstack-baremetal-chassis-list
 command line option, [14](#)
 openstack-baremetal-conductor-list
 command line option, [17](#)
 openstack-baremetal-deploy-template-list
 command line option, [21](#)
 openstack-baremetal-node-list
 command line option, [45](#)
 openstack-baremetal-port-group-list
 command line option, [71](#)
 openstack-baremetal-port-list
 command line option, [75](#)
 openstack-baremetal-volume-connector-list
 command line option, [81](#)
 openstack-baremetal-volume-target-list
 command line option, [85](#)
--sort-ascending
 openstack-baremetal-allocation-list
 command line option, [10](#)
 openstack-baremetal-chassis-list
 command line option, [14](#)
 openstack-baremetal-conductor-list
 command line option, [17](#)
 openstack-baremetal-deploy-template-list
 command line option, [21](#)
 openstack-baremetal-driver-list
 command line option, [24](#)
 openstack-baremetal-driver-passthru-list
 command line option, [26](#)
 openstack-baremetal-driver-property-list
 command line option, [27](#)
 openstack-baremetal-driver-raid-property-list
 command line option, [28](#)
 openstack-baremetal-node-bios-setting-list
 command line option, [31](#)
 openstack-baremetal-node-firmware-list
 command line option, [41](#)
 openstack-baremetal-node-history-list
 command line option, [43](#)
 openstack-baremetal-node-list
 command line option, [45](#)
 openstack-baremetal-node-passthru-list
 command line option, [49](#)
 openstack-baremetal-node-trait-list
 command line option, [59](#)
 openstack-baremetal-node-validate
 command line option, [64](#)
 openstack-baremetal-node-vif-list
 command line option, [66](#)
--sort-column
 openstack-baremetal-allocation-list

command line option, 10
 openstack-baremetal-chassis-list
 command line option, 14
 openstack-baremetal-conductor-list
 command line option, 17
 openstack-baremetal-deploy-template-list
 command line option, 21
 openstack-baremetal-driver-list
 command line option, 24
 openstack-baremetal-driver-passthru-list
 command line option, 26
 openstack-baremetal-driver-property-list
 command line option, 27
 openstack-baremetal-driver-raid-property-list
 command line option, 28
 openstack-baremetal-node-bios-setting-list
 command line option, 31
 openstack-baremetal-node-firmware-list
 command line option, 41
 openstack-baremetal-node-history-list
 command line option, 43
 openstack-baremetal-node-list
 command line option, 45
 openstack-baremetal-node-passthru-list
 command line option, 49
 openstack-baremetal-node-trait-list
 command line option, 59
 openstack-baremetal-node-validate
 command line option, 64
 openstack-baremetal-node-vif-list
 command line option, 66
 openstack-baremetal-port-group-list
 command line option, 71
 openstack-baremetal-port-list
 command line option, 75
 openstack-baremetal-volume-connector-list
 command line option, 81
 openstack-baremetal-volume-target-list
 command line option, 85
 --sort-descending
 openstack-baremetal-allocation-list
 command line option, 10
 openstack-baremetal-chassis-list
 command line option, 14
 openstack-baremetal-conductor-list
 command line option, 17
 openstack-baremetal-deploy-template-list
 command line option, 21
 openstack-baremetal-driver-list
 command line option, 24
 openstack-baremetal-driver-passthru-list
 command line option, 26
 openstack-baremetal-driver-property-list
 command line option, 27
 openstack-baremetal-driver-raid-property-list
 command line option, 28
 openstack-baremetal-node-bios-setting-list
 command line option, 31
 openstack-baremetal-node-firmware-list
 command line option, 41
 openstack-baremetal-node-history-list
 command line option, 43
 openstack-baremetal-node-list
 command line option, 45
 openstack-baremetal-node-passthru-list
 command line option, 49
 openstack-baremetal-node-trait-list
 command line option, 59
 openstack-baremetal-node-validate
 command line option, 64
 openstack-baremetal-node-vif-list
 command line option, 66
 openstack-baremetal-port-group-list
 command line option, 71
 openstack-baremetal-port-list
 command line option, 75
 openstack-baremetal-volume-connector-list
 command line option, 81
 openstack-baremetal-volume-target-list
 command line option, 85
 --state
 openstack-baremetal-allocation-list
 command line option, 10
 --steps
 openstack-baremetal-deploy-template-create
 command line option, 20
 openstack-baremetal-deploy-template-set
 command line option, 22
 storage-interface
 openstack-baremetal-node-create
 command line option, 38
 openstack-baremetal-node-set
 command line option, 55
 openstack-baremetal-node-unset
 command line option, 62
 --support-standalone-ports
 openstack-baremetal-port-group-create
 command line option, 69
 openstack-baremetal-port-group-set
 command line option, 72
 --supported
 openstack-baremetal-node-boot-device-show

command line option, 33

--target-raid-config

- openstack-baremetal-node-set
command line option, 56
- openstack-baremetal-node-unset
command line option, 61

--trait

- openstack-baremetal-allocation-create
command line option, 8

--type

- openstack-baremetal-driver-list
command line option, 24
- openstack-baremetal-volume-connector-create
command line option, 79
- openstack-baremetal-volume-connector-set
command line option, 81
- openstack-baremetal-volume-target-create
command line option, 84
- openstack-baremetal-volume-target-set
command line option, 86

--unassociated

- openstack-baremetal-node-list
command line option, 46

--unsharded

- openstack-baremetal-node-list
command line option, 46

--unsupport-standalone-ports

- openstack-baremetal-port-group-create
command line option, 69
- openstack-baremetal-port-group-set
command line option, 72

--uuid

- openstack-baremetal-allocation-create
command line option, 8
- openstack-baremetal-chassis-create
command line option, 13
- openstack-baremetal-deploy-template-create
command line option, 20
- openstack-baremetal-node-create
command line option, 37
- openstack-baremetal-port-create
command line option, 67
- openstack-baremetal-port-group-create
command line option, 69
- openstack-baremetal-volume-connector-create
command line option, 79
- openstack-baremetal-volume-target-create
command line option, 84

--vendor

- openstack-baremetal-port-create
command line option, 67
- openstack-baremetal-port-set
command line option, 76
- openstack-baremetal-port-unset
command line option, 78

--vendor-interface

- openstack-baremetal-node-create
command line option, 39
- openstack-baremetal-node-set
command line option, 56
- openstack-baremetal-node-unset
command line option, 62

--vif-info

- openstack-baremetal-node-vif-attach
command line option, 64

volume-id

- openstack-baremetal-volume-target-create
command line option, 84
- openstack-baremetal-volume-target-set
command line option, 86

--wait

- openstack-baremetal-allocation-create
command line option, 8
- openstack-baremetal-node-adopt
command line option, 30
- openstack-baremetal-node-clean
command line option, 34
- openstack-baremetal-node-deploy
command line option, 40
- openstack-baremetal-node-inspect
command line option, 43
- openstack-baremetal-node-manage
command line option, 47
- openstack-baremetal-node-provide
command line option, 50
- openstack-baremetal-node-rebuild
command line option, 51
- openstack-baremetal-node-rescue
command line option, 52
- openstack-baremetal-node-service
command line option, 53
- openstack-baremetal-node-undeploy
command line option, 60
- openstack-baremetal-node-unrescue
command line option, 60
- openstack-baremetal-allocation-create
command line option, 8
- openstack-baremetal-allocation-list
command line option, 9
- openstack-baremetal-allocation-show
command line option, 11

openstack-baremetal-chassis-create	command line option, 57
openstack-baremetal-chassis-list	command line option, 58
openstack-baremetal-chassis-show	command line option, 64
openstack-baremetal-conductor-list	command line option, 65
openstack-baremetal-conductor-show	command line option, 67
openstack-baremetal-deploy-template-create	command line option, 68
openstack-baremetal-deploy-template-list	command line option, 70
openstack-baremetal-deploy-template-show	command line option, 73
openstack-baremetal-driver-list	command line option, 74
openstack-baremetal-driver-passthru-call	command line option, 77
openstack-baremetal-driver-passthru-list	command line option, 79
openstack-baremetal-driver-property-list	command line option, 80
openstack-baremetal-driver-raid-property-list	command line option, 82
openstack-baremetal-driver-show	command line option, 83
openstack-baremetal-node-bios-setting-list	command line option, 85
openstack-baremetal-node-bios-setting-show	command line option, 87
openstack-baremetal-node-boot-device-show	-f
openstack-baremetal-node-children-list	openstack-baremetal-allocation-create
openstack-baremetal-node-console-show	command line option, 8
openstack-baremetal-node-create	openstack-baremetal-allocation-list
openstack-baremetal-node-firmware-list	command line option, 9
openstack-baremetal-node-history-get	openstack-baremetal-allocation-show
openstack-baremetal-node-history-list	command line option, 11
openstack-baremetal-node-list	openstack-baremetal-chassis-create
openstack-baremetal-node-passthru-list	command line option, 12
openstack-baremetal-node-show	openstack-baremetal-chassis-list
	command line option, 14
	openstack-baremetal-chassis-show
	command line option, 15
	openstack-baremetal-conductor-list
	command line option, 17
	openstack-baremetal-conductor-show
	command line option, 18
	openstack-baremetal-deploy-template-create
	command line option, 19
	openstack-baremetal-deploy-template-list

- command line option, 20
 - openstack-baremetal-deploy-template-show
 - command line option, 22
 - openstack-baremetal-driver-list
 - command line option, 23
 - openstack-baremetal-driver-passthru-call
 - command line option, 24
 - openstack-baremetal-driver-passthru-list
 - command line option, 25
 - openstack-baremetal-driver-property-list
 - command line option, 26
 - openstack-baremetal-driver-raid-property-list
 - command line option, 27
 - openstack-baremetal-driver-show
 - command line option, 28
 - openstack-baremetal-node-bios-setting-list
 - command line option, 30
 - openstack-baremetal-node-bios-setting-show
 - command line option, 31
 - openstack-baremetal-node-boot-device-show
 - command line option, 32
 - openstack-baremetal-node-children-list
 - command line option, 34
 - openstack-baremetal-node-console-show
 - command line option, 35
 - openstack-baremetal-node-create
 - command line option, 37
 - openstack-baremetal-node-firmware-list
 - command line option, 40
 - openstack-baremetal-node-history-get
 - command line option, 41
 - openstack-baremetal-node-history-list
 - command line option, 42
 - openstack-baremetal-node-list
 - command line option, 45
 - openstack-baremetal-node-passthru-list
 - command line option, 48
 - openstack-baremetal-node-show
 - command line option, 57
 - openstack-baremetal-node-trait-list
 - command line option, 58
 - openstack-baremetal-node-validate
 - command line option, 63
 - openstack-baremetal-node-vif-list
 - command line option, 65
 - openstack-baremetal-port-create
 - command line option, 66
 - openstack-baremetal-port-group-create
 - command line option, 68
 - openstack-baremetal-port-group-list
 - command line option, 70
 - openstack-baremetal-port-group-show
 - command line option, 72
 - openstack-baremetal-port-list
 - command line option, 74
 - openstack-baremetal-port-show
 - command line option, 77
 - openstack-baremetal-volume-connector-create
 - command line option, 79
 - openstack-baremetal-volume-connector-list
 - command line option, 80
 - openstack-baremetal-volume-connector-show
 - command line option, 82
 - openstack-baremetal-volume-target-create
 - command line option, 83
 - openstack-baremetal-volume-target-list
 - command line option, 85
 - openstack-baremetal-volume-target-show
 - command line option, 87
- A**
- AbortBaremetalNode (class in *ironic-client.osc.v1.baremetal_node*), 135
 - add_trait() (*ironicclient.v1.node.NodeManager* method), 183
 - add_trait() (*ironic-client.v1.runbook.RunbookManager* method), 203
 - address
 - openstack-baremetal-port-create
 - command line option, 68
 - AddTraitBaremetalNode (class in *ironic-client.osc.v1.baremetal_node*), 135
 - AddTraitBaremetalRunbook (class in *ironic-client.osc.v1.baremetal_runbook*), 158
 - AdoptBaremetalNode (class in *ironic-client.osc.v1.baremetal_node*), 136
 - 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 *ironicclient.v1.allocation*), 170
 - AllocationManager (class in *ironic-client.v1.allocation*), 170
 - AmbiguousAuthSystem (in module *ironic-client.exc*), 209
 - AmbiguousAuthSystem, 209
 - AmbiguousEndpoints, 93

- `api_version_select_state` (*ironic-client.common.http.VersionNegotiationMixin* attribute), 114
- `App` (class in *ironicclient.shell*), 210
- `app` (*ironicclient.osc.command.Command* attribute), 169
- `args_array_to_dict()` (in module *ironic-client.common.utils*), 115
- `args_array_to_patch()` (in module *ironic-client.common.utils*), 115
- `AuthorizationFailure`, 94
- `AuthPluginOptionsMissing`, 93
- `AuthSystemNotFound`, 94
- ## B
- `BadGateway`, 94
- `BadRequest`, 94
- `baremetal` (*ironicclient.shell.ClientManager* property), 210
- `BIOSSettingShowBaremetalNode` (class in *ironicclient.osc.v1.baremetal_node*), 136
- `bool_argument_value()` (in module *ironic-client.common.utils*), 115
- `boot_mode`
- `openstack-baremetal-node-boot-mode-set` command line option, 33
- `BootdeviceSetBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 136
- `BootdeviceShowBaremetalNode` (class in *ironicclient.osc.v1.baremetal_node*), 137
- `BootmodeSetBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 137
- `build_option_parser()` (in module *ironic-client.osc.plugin*), 170
- `build_option_parser()` (*ironicclient.shell.App* method), 210
- ## 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*), 172
- `ChassisManager` (class in *ironic-client.v1.chassis*), 172
- `check_api_version_support()` (in module *ironicclient.common.utils*), 116
- `check_empty_arg()` (in module *ironic-client.common.utils*), 116
- `check_for_invalid_fields()` (in module *ironicclient.common.utils*), 116
- `CleanBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 138
- `Client` (class in *ironicclient.v1.client*), 174
- `client` (*ironicclient.common.apiclient.base.ManagerProtocol* attribute), 93
- `Client()` (in module *ironicclient.client*), 208
- `ClientException`, 95
- `ClientManager` (class in *ironicclient.shell*), 210
- `Command` (class in *ironicclient.osc.command*), 169
- `CommandError`, 95
- `CommandManager` (class in *ironicclient.shell*), 210
- `common_filters()` (in module *ironic-client.common.utils*), 116
- `common_params_for_list()` (in module *ironic-client.common.utils*), 117
- `conductor`
- `openstack-baremetal-conductor-show` command line option, 18
- `Conductor` (class in *ironicclient.v1.conductor*), 175
- `ConductorManager` (class in *ironic-client.v1.conductor*), 175
- `Conflict`, 95
- `ConnectionError`, 95
- `ConnectionRefused`, 95
- `ConsoleDisableBaremetalNode` (class in *ironicclient.osc.v1.baremetal_node*), 138
- `ConsoleEnableBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 138
- `ConsoleShowBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 139
- `convert_list_props_to_comma_separated()` (in module *ironicclient.common.utils*), 117
- `create()` (*ironic-client.common.base.CreateManager* method), 113
- `create_chassis()` (in module *ironic-client.v1.create_resources*), 176
- `create_nodes()` (in module *ironic-client.v1.create_resources*), 176
- `create_portgroups()` (in module *ironic-client.v1.create_resources*), 176
- `create_ports()` (in module *ironic-client.v1.create_resources*), 177
- `create_resources()` (in module *ironic-client.v1.create_resources*), 177

- `create_single_chassis()` (in module `ironic-client.v1.create_resources`), 177
- `create_single_handler()` (in module `ironic-client.v1.create_resources`), 177
- `create_single_node()` (in module `ironic-client.v1.create_resources`), 178
- `create_single_port()` (in module `ironic-client.v1.create_resources`), 178
- `create_single_portgroup()` (in module `ironicclient.v1.create_resources`), 178
- `CreateBaremetal` (class in `ironic-client.osc.v1.baremetal_create`), 125
- `CreateBaremetalAllocation` (class in `ironic-client.osc.v1.baremetal_allocation`), 119
- `CreateBaremetalChassis` (class in `ironic-client.osc.v1.baremetal_chassis`), 122
- `CreateBaremetalDeployTemplate` (class in `ironic-client.osc.v1.baremetal_deploy_template`), 126
- `CreateBaremetalInspectionRule` (class in `ironic-client.osc.v1.baremetal_inspection_rule`), 132
- `CreateBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 139
- `CreateBaremetalPort` (class in `ironic-client.osc.v1.baremetal_port`), 154
- `CreateBaremetalPortGroup` (class in `ironic-client.osc.v1.baremetal_portgroup`), 156
- `CreateBaremetalRunbook` (class in `ironic-client.osc.v1.baremetal_runbook`), 159
- `CreateBaremetalVolumeConnector` (class in `ironic-client.osc.v1.baremetal_volume_connector`), 162
- `CreateBaremetalVolumeTarget` (class in `ironic-client.osc.v1.baremetal_volume_target`), 165
- `CreateManager` (class in `ironic-client.common.base`), 113
- `current_api_version` (`ironic-client.v1.client.Client` property), 174
- D**
- `delete()` (`ironic-client.v1.allocation.AllocationManager` method), 170
- `delete()` (`ironic-client.v1.chassis.ChassisManager` method), 172
- `delete()` (`ironic-client.v1.deploy_template.DeployTemplateManager` method), 179
- `delete()` (`ironicclient.v1.driver.DriverManager` method), 180
- `delete()` (`ironic-client.v1.inspection_rule.InspectionRuleManager` method), 182
- `delete()` (`ironicclient.v1.node.NodeManager` method), 183
- `delete()` (`ironicclient.v1.port.PortManager` method), 197
- `delete()` (`ironic-client.v1.portgroup.PortgroupManager` method), 198
- `delete()` (`ironic-client.v1.runbook.RunbookManager` method), 203
- `delete()` (`ironic-client.v1.volume_connector.VolumeConnectorManager` method), 206
- `delete()` (`ironic-client.v1.volume_target.VolumeTargetManager` method), 207
- `DeleteBaremetalAllocation` (class in `ironic-client.osc.v1.baremetal_allocation`), 119
- `DeleteBaremetalChassis` (class in `ironic-client.osc.v1.baremetal_chassis`), 122
- `DeleteBaremetalDeployTemplate` (class in `ironic-client.osc.v1.baremetal_deploy_template`), 126
- `DeleteBaremetalInspectionRule` (class in `ironic-client.osc.v1.baremetal_inspection_rule`), 132
- `DeleteBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 140
- `DeleteBaremetalPort` (class in `ironic-client.osc.v1.baremetal_port`), 154
- `DeleteBaremetalPortGroup` (class in `ironic-client.osc.v1.baremetal_portgroup`), 156
- `DeleteBaremetalRunbook` (class in `ironic-client.osc.v1.baremetal_runbook`), 159
- `DeleteBaremetalVolumeConnector` (class in `ironic-client.osc.v1.baremetal_volume_connector`), 163
- `DeleteBaremetalVolumeTarget` (class in `ironic-`

`client.osc.v1.baremetal_volume_target`), 166

`DeployBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 140

`DeployTemplate` (class in `ironic-client.v1.deploy_template`), 179

`DeployTemplateManager` (class in `ironic-client.v1.deploy_template`), 179

`detail` (`ironicclient.common.utils.ListCommandArgs` attribute), 115

`device`

- `openstack-baremetal-node-boot-device-set` command line option, 32

`driver`

- `openstack-baremetal-driver-passthru-get` command line option, 25
- `openstack-baremetal-driver-passthru-list` command line option, 26
- `openstack-baremetal-driver-property-list` command line option, 27
- `openstack-baremetal-driver-raid-property-list` command line option, 28
- `openstack-baremetal-driver-show` command line option, 29

`Driver` (class in `ironicclient.v1.driver`), 180

`DriverManager` (class in `ironicclient.v1.driver`), 180

E

`EndpointException`, 95

`EndpointNotFound`, 95

`event`

- `openstack-baremetal-node-history-get` command line option, 42

`Event` (class in `ironicclient.v1.events`), 182

`EventManager` (class in `ironicclient.v1.events`), 182

`ExpectationFailed`, 95

F

`fields` (`ironicclient.common.utils.ListCommandArgs` attribute), 115

`FIELDS` (`ironicclient.v1.resource_fields.Resource` attribute), 201

`fields` (`ironicclient.v1.resource_fields.Resource` property), 203

`file`

- `openstack-baremetal-create` command line option, 19

`Forbidden`, 96

`format_hash()` (in module `ironic-client.common.utils`), 117

`from_response()` (in module `ironic-client.common.apiclient.exceptions`), 112

`from_response()` (in module `ironicclient.exc`), 209

G

`GatewayTimeout`, 96

`get()` (`ironicclient.common.apiclient.base.Resource` method), 93

`get()` (`ironicclient.v1.allocation.AllocationManager` method), 170

`get()` (`ironicclient.v1.chassis.ChassisManager` method), 172

`get()` (`ironicclient.v1.conductor.ConductorManager` method), 175

`get()` (`ironicclient.v1.deploy_template.DeployTemplateManager` method), 179

`get()` (`ironicclient.v1.driver.DriverManager` method), 180

`get()` (`ironicclient.v1.inspection_rule.InspectionRuleManager` method), 182

`get()` (`ironicclient.v1.node.NodeManager` method), 183

`get()` (`ironicclient.v1.port.PortManager` method), 197

`get()` (`ironicclient.v1.portgroup.PortgroupManager` method), 198

`get()` (`ironicclient.v1.runbook.RunbookManager` method), 203

`get()` (`ironicclient.v1.volume_connector.VolumeConnectorManager` method), 206

`get()` (`ironicclient.v1.volume_target.VolumeTargetManager` method), 207

`get_bios_setting()` (`ironic-client.v1.node.NodeManager` method), 184

`get_boot_device()` (`ironic-client.v1.node.NodeManager` method), 184

`get_by_address()` (`ironic-client.v1.port.PortManager` method), 197

`get_by_address()` (`ironic-client.v1.portgroup.PortgroupManager` method), 199

`get_by_instance_uuid()` (`ironic-client.v1.node.NodeManager` method), 184

`get_client()` (in module `ironicclient.client`), 208

`get_console()` (`ironic-`

<code>client.v1.node.NodeManager</code> method), 184	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_conductor.ListBaremetalConducto</code> method), 125
<code>get_from_stdin()</code> (in module <code>ironic-</code> <code>client.common.utils</code>), 117	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_conductor.ShowBaremetalCondu</code> method), 125
<code>get_history_event()</code> (ironic- <code>client.v1.node.NodeManager</code> method), 184	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_create.CreateBaremetal</code> method), 125
<code>get_history_list()</code> (ironic- <code>client.v1.node.NodeManager</code> method), 184	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.CreateBaremeta</code> method), 126
<code>get_inventory()</code> (ironic- <code>client.v1.node.NodeManager</code> method), 185	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.DeleteBaremeta</code> method), 127
<code>get_json_data()</code> (in module <code>ironic-</code> <code>client.common.utils</code>), 118	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.CreateBaremetalAllocati</code> method), 119
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.DeleteBaremetalAllocati</code> method), 120	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.DeleteBaremetalAllocati</code> method), 128
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.ListBaremetalAllocati</code> method), 120	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.ListBaremetalDe</code> method), 127
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.SetBaremetalAllocati</code> method), 121	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.SetBaremetalDe</code> method), 128
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.ShowBaremetalAllocati</code> method), 121	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.ShowBaremetalL</code> method), 128
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.UnsetBaremetalAllocati</code> method), 122	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_deploy_template.UnsetBaremetalL</code> method), 129
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.ShowBaremetalAllocati</code> method), 121	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.ListBaremetalDriver</code> method), 129
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_allocation.UnsetBaremetalAllocati</code> method), 122	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.ListBaremetalDriverPrope</code> method), 130
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.CreateBaremetalChassis</code> method), 122	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.ListBaremetalDriverRaidP</code> method), 130
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.DeleteBaremetalChassis</code> method), 123	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.PassthruCallBaremetalDri</code> method), 131
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.ListBaremetalChassis</code> method), 123	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.PassthruListBaremetalDri</code> method), 131
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.SetBaremetalChassis</code> method), 123	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_driver.ShowBaremetalDriver</code> method), 131
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.ShowBaremetalChassis</code> method), 124	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_inspection_rule.CreateBaremetal</code> method), 132
<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_chassis.UnsetBaremetalChassis</code> method), 124	<code>get_parser()</code> (ironic- <code>client.osc.v1.baremetal_inspection_rule.DeleteBaremetal</code> method), 133

<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_inspection_rule.ListBaremetalInspectionRule</code>		<code>client.osc.v1.baremetal_node.ListBaremetalNode</code>	
<code>method), 133</code>		<code>method), 141</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_inspection_rule.SetBaremetalInspectionRule</code>		<code>client.osc.v1.baremetal_node.ListBIOSSettingBaremetalN</code>	
<code>method), 134</code>		<code>method), 141</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_inspection_rule.ShowBaremetalInspectionRule</code>		<code>client.osc.v1.baremetal_node.ListFirmwareComponentBa</code>	
<code>method), 134</code>		<code>method), 142</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_inspection_rule.UnsetBaremetalInspectionRule</code>		<code>client.osc.v1.baremetal_node.ListTraitsBaremetalNode</code>	
<code>method), 135</code>		<code>method), 142</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.AddTraitBaremetalNode</code>		<code>client.osc.v1.baremetal_node.MaintenanceSetBaremetalN</code>	
<code>method), 136</code>		<code>method), 143</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode</code>		<code>client.osc.v1.baremetal_node.MaintenanceUnsetBaremeta</code>	
<code>method), 136</code>		<code>method), 143</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.BootdeviceSetBaremetalNode</code>		<code>client.osc.v1.baremetal_node.NodeChildrenList</code>	
<code>method), 137</code>		<code>method), 144</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.BootdeviceShowBaremetalNode</code>		<code>client.osc.v1.baremetal_node.NodeHistoryEventGet</code>	
<code>method), 137</code>		<code>method), 144</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.BootmodeSetBaremetalNode</code>		<code>client.osc.v1.baremetal_node.NodeHistoryList</code>	
<code>method), 137</code>		<code>method), 144</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.CleanBaremetalNode</code>		<code>client.osc.v1.baremetal_node.NodeInventorySave</code>	
<code>method), 138</code>		<code>method), 145</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.ConsoleDisableBaremetalNode</code>		<code>client.osc.v1.baremetal_node.PassthruCallBaremetalNode</code>	
<code>method), 138</code>		<code>method), 145</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.ConsoleEnableBaremetalNode</code>		<code>client.osc.v1.baremetal_node.PassthruListBaremetalNode</code>	
<code>method), 139</code>		<code>method), 146</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.ConsoleShowBaremetalNode</code>		<code>client.osc.v1.baremetal_node.PowerBaremetalNode</code>	
<code>method), 139</code>		<code>method), 146</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.CreateBaremetalNode</code>		<code>client.osc.v1.baremetal_node.PowerOffBaremetalNode</code>	
<code>method), 139</code>		<code>method), 146</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.DeleteBaremetalNode</code>		<code>client.osc.v1.baremetal_node.ProvisionStateBaremetalNo</code>	
<code>method), 140</code>		<code>method), 147</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.DeployBaremetalNode</code>		<code>client.osc.v1.baremetal_node.ProvisionStateWithWait</code>	
<code>method), 140</code>		<code>method), 148</code>	
<code>get_parser()</code>	(<code>ironic-</code>	<code>get_parser()</code>	(<code>ironic-</code>
<code>client.osc.v1.baremetal_node.InjectNmiBaremetalNode</code>		<code>client.osc.v1.baremetal_node.RebootBaremetalNode</code>	
<code>method), 140</code>		<code>method), 148</code>	

<code>get_parser()</code> <i>client.osc.v1.baremetal_node.RebuildBaremetalNode</i> <i>method), 148</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</i> <i>method), 149</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.ShowBaremetalPort</i> <i>method), 155</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</i> <i>method), 149</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.SecurebootOffBaremetalNode</i> <i>method), 149</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.UnsetBaremetalPort</i> <i>method), 155</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.RescueBaremetalNode</i> <i>method), 149</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.SecurebootOnBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.CreateBaremetalPortG</i> <i>method), 156</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.SecurebootOffBaremetalNode</i> <i>method), 149</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.SecurebootOnBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.DeleteBaremetalPortG</i> <i>method), 156</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.SecurebootOnBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.ServiceBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.ListBaremetalPortGrou</i> <i>method), 157</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.ServiceBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.SetBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.SetBaremetalPortGrou</i> <i>method), 157</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.SetBaremetalNode</i> <i>method), 150</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.ShowBaremetalNode</i> <i>method), 151</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.ShowBaremetalPortGr</i> <i>method), 157</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.ShowBaremetalNode</i> <i>method), 151</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.UnsetBaremetalNode</i> <i>method), 152</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_portgroup.UnsetBaremetalPortGr</i> <i>method), 158</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.UnsetBaremetalNode</i> <i>method), 152</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.ValidateBaremetalNode</i> <i>method), 152</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.AddTraitBaremetalRunb</i> <i>method), 158</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.ValidateBaremetalNode</i> <i>method), 152</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifAttachBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.CreateBaremetalRunboo</i> <i>method), 159</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifAttachBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifDetachBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.DeleteBaremetalRunboo</i> <i>method), 159</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifDetachBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifListBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.ListBaremetalRunbook</i> <i>method), 160</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_node.VifListBaremetalNode</i> <i>method), 153</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.CreateBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.ListTraitsBaremetalRunb</i> <i>method), 160</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_port.CreateBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.DeleteBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.RemoveTraitBaremetalR</i> <i>method), 160</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_port.DeleteBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.ListBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.SetBaremetalRunbook</i> <i>method), 161</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_port.ListBaremetalPort</i> <i>method), 154</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_port.SetBaremetalPort</i> <i>method), 155</i>	(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.ShowBaremetalRunbook</i> <i>method), 161</i>
<code>get_parser()</code> <i>client.osc.v1.baremetal_port.SetBaremetalPort</i> <i>method), 155</i>		(<i>ironic-</i> <code>get_parser()</code> <i>client.osc.v1.baremetal_runbook.UnsetBaremetalRunbook</i> <i>method), 162</i>

get_parser() (ironic- client.v1.node.NodeManager method),
client.osc.v1.baremetal_volume_connector.CreateBaremetalVolumeConnector
method), 162
get_parser() (ironic- 113
client.osc.v1.baremetal_volume_connector.DeleteBaremetalVolumeConnector
method), 163
get_parser() (ironic- H
client.osc.v1.baremetal_volume_connector.HandleJsonArg() (in module ironic-
method), 163 client.common.utils), 118
get_parser() (ironic- handle_json_or_file_arg() (in module iron-
client.osc.v1.baremetal_volume_connector.SetBaremetalVolumeConnatils), 118
method), 164 HashColumn (class in ironicclient.common.utils),
get_parser() (ironic- 115
client.osc.v1.baremetal_volume_connector.StopBaremetalVolumeConnector (ironic-
method), 164 client.common.apiclient.exceptions.BadGateway
get_parser() (ironic- attribute), 94
client.osc.v1.baremetal_volume_connector.UpdateBaremetalVolumeConnector (ironic-
method), 165 client.common.apiclient.exceptions.BadRequest
get_parser() (ironic- attribute), 94
client.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget (ironic-
method), 166 client.common.apiclient.exceptions.Conflict
get_parser() (ironic- attribute), 95
client.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget (ironic-
method), 166 client.common.apiclient.exceptions.ExpectationFailed
get_parser() (ironic- attribute), 96
client.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget (ironic-
method), 167 client.common.apiclient.exceptions.Forbidden
get_parser() (ironic- attribute), 96
client.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget (ironic-
method), 167 client.common.apiclient.exceptions.GatewayTimeout
get_parser() (ironic- attribute), 97
client.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget (ironic-
method), 168 client.common.apiclient.exceptions.Gone
get_parser() (ironic- attribute), 97
client.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget (ironic-
method), 168 client.common.apiclient.exceptions.HttpError
get_server() (in module ironic- attribute), 99
client.common.http), 114 http_status (ironic-
get_supported_boot_devices() (ironic- client.common.apiclient.exceptions.HttpNotImplemented
client.v1.node.NodeManager method), attribute), 99
185 http_status (ironic-
get_traits() (ironic- client.common.apiclient.exceptions.HttpVersionNotSupport
client.v1.node.NodeManager method), attribute), 101
185 http_status (ironic-
get_traits() (ironic- client.common.apiclient.exceptions.InternalServerError
client.v1.runbook.RunbookManager attribute), 102
method), 203 http_status (ironic-
get_vendor_passthru_methods() (ironic- client.common.apiclient.exceptions.LengthRequired
client.v1.driver.DriverManager method), attribute), 102
180 http_status (ironic-
get_vendor_passthru_methods() (ironic- client.common.apiclient.exceptions.MethodNotAllowed

attribute), 103
 http_status (ironic- human_id (ironic-
 client.common.apiclient.exceptions.MultipleChoices client.common.apiclient.base.Resource
 attribute), 103 property), 93
 http_status (ironic- human_readable() (ironic-
 client.common.apiclient.exceptions.NotAcceptable client.common.utils.HashColumn
 attribute), 104 method), 115
 http_status (ironic-
 client.common.apiclient.exceptions.NotFound
 attribute), 104 id
 http_status (ironic- openstack-baremetal-allocation-show
 client.common.apiclient.exceptions.PaymentRequired command line option, 12
 attribute), 105 openstack-baremetal-port-group-show
 http_status (ironic- command line option, 73
 client.common.apiclient.exceptions.PreconditionFailed openstack-baremetal-port-show
 attribute), 105 command line option, 77
 http_status (ironic- openstack-baremetal-volume-connector-show
 client.common.apiclient.exceptions.ProxyAuthenticationRequired command line option, 82
 attribute), 107 openstack-baremetal-volume-target-show
 http_status (ironic- command line option, 87
 client.common.apiclient.exceptions.RequestEntityTooLarge (ironicclient.shell.App
 attribute), 109 method), 210
 http_status (ironic- inject_rmi() (ironic-
 client.common.apiclient.exceptions.RequestEntityTooLarge client.osc.v1.node.NodeManager method),
 attribute), 107 185
 http_status (ironic- InjectNmiBaremetalNode (class in ironic-
 client.common.apiclient.exceptions.RequestTimeout client.osc.v1.baremetal_node), 140
 attribute), 107 InspectBaremetalNode (class in ironic-
 http_status (ironic- client.osc.v1.baremetal_node), 141
 client.common.apiclient.exceptions.RequestTimeout InspectingRule (class in ironic-
 attribute), 108 client.v1.inspection_rule), 182
 http_status (ironic- InspectionRuleManager (class in ironic-
 client.common.apiclient.exceptions.ServiceUnavailable client.v1.inspection_rule), 182
 attribute), 110 InternalServerError, 101
 http_status (ironic- InvalidAttribute, 209
 client.common.apiclient.exceptions.Unauthorized ironicclient
 attribute), 110 module, 210
 http_status (ironic- ironicclient.client
 client.common.apiclient.exceptions.UnprocessableEntity module, 208
 attribute), 111 ironicclient.common
 http_status (ironic- module, 119
 client.common.apiclient.exceptions.UnsupportedMediaTypes ironicclient.common.apiclient
 attribute), 112 module, 113
 HTTPClientError, 97 ironicclient.common.apiclient.base
 HttpError, 98 module, 93
 HttpNotImplemented, 99 ironicclient.common.apiclient.exceptions
 HTTPRedirection, 98 module, 93
 HttpServerError, 99 ironicclient.common.base
 HttpVersionNotSupported, 100 module, 113
 HUMAN_ID (ironic- ironicclient.common.filecache
 client.common.apiclient.base.Resource module, 114

`ironicclient.common.http`
 module, 114
`ironicclient.common.i18n`
 module, 115
`ironicclient.common.utils`
 module, 115
`ironicclient.exc`
 module, 209
`ironicclient.osc`
 module, 170
`ironicclient.osc.command`
 module, 169
`ironicclient.osc.plugin`
 module, 169
`ironicclient.osc.v1`
 module, 169
`ironicclient.osc.v1.baremetal_allocation`
 module, 119
`ironicclient.osc.v1.baremetal_chassis`
 module, 122
`ironicclient.osc.v1.baremetal_conductor`
 module, 124
`ironicclient.osc.v1.baremetal_create`
 module, 125
`ironicclient.osc.v1.baremetal_deploy_template`
 module, 126
`ironicclient.osc.v1.baremetal_driver`
 module, 129
`ironicclient.osc.v1.baremetal_inspection_rule`
 module, 132
`ironicclient.osc.v1.baremetal_node`
 module, 135
`ironicclient.osc.v1.baremetal_port`
 module, 154
`ironicclient.osc.v1.baremetal_portgroup`
 module, 156
`ironicclient.osc.v1.baremetal_runbook`
 module, 158
`ironicclient.osc.v1.baremetal_shard`
 module, 162
`ironicclient.osc.v1.baremetal_volume_connector`
 module, 162
`ironicclient.osc.v1.baremetal_volume_target`
 module, 165
`ironicclient.shell`
 module, 210
`ironicclient.v1`
 module, 208
`ironicclient.v1.allocation`
 module, 170
`ironicclient.v1.chassis`
 module, 172
`ironicclient.v1.client`
 module, 174
`ironicclient.v1.conductor`
 module, 175
`ironicclient.v1.create_resources`
 module, 176
`ironicclient.v1.deploy_template`
 module, 179
`ironicclient.v1.driver`
 module, 180
`ironicclient.v1.events`
 module, 182
`ironicclient.v1.inspection_rule`
 module, 182
`ironicclient.v1.node`
 module, 183
`ironicclient.v1.port`
 module, 197
`ironicclient.v1.portgroup`
 module, 198
`ironicclient.v1.resource_fields`
 module, 201
`ironicclient.v1.runbook`
 module, 203
`ironicclient.v1.shard`
 module, 205
`ironicclient.v1.utils`
 module, 205
`ironicclient.v1.volume_connector`
 module, 205
`ironicclient.v1.volume_target`
 module, 207
`is_api_version_negotiated` (*ironic-client.v1.client.Client* property), 175
`is_loaded()` (*ironic-client.common.apiclient.base.Resource* method), 93

J

`json_request()` (*ironic-client.common.http.SessionClient* method), 114

K

`key_value_pairs_to_dict()` (*in module ironicclient.common.utils*), 118

L

`labels` (*ironicclient.v1.resource_fields.Resource* property), 203
`LengthRequired`, 102

<code>license</code>		<code>list_volume_targets()</code>	(<i>ironic-client.v1.node.NodeManager</i> method),
<code> agreement</code>	91		
<code>limit</code>	(<i>ironicclient.common.utils.ListCommandArgs</i> attribute), 115		
<code>list()</code>	(<i>ironicclient.v1.allocation.AllocationManager</i> method), 170	<code>ListBaremetalAllocation</code>	(class in <i>ironic-client.osc.v1.baremetal_allocation</i>), 120
<code>list()</code>	(<i>ironicclient.v1.chassis.ChassisManager</i> method), 172	<code>ListBaremetalChassis</code>	(class in <i>ironic-client.osc.v1.baremetal_chassis</i>), 123
<code>list()</code>	(<i>ironicclient.v1.conductor.ConductorManager</i> method), 175	<code>ListBaremetalConductor</code>	(class in <i>ironic-client.osc.v1.baremetal_conductor</i>), 124
<code>list()</code>	(<i>ironicclient.v1.deploy_template.DeployTemplateManager</i> method), 179	<code>ListBaremetalDeployTemplate</code>	(class in <i>ironic-client.osc.v1.baremetal_deploy_template</i>), 127
<code>list()</code>	(<i>ironicclient.v1.driver.DriverManager</i> method), 180	<code>ListBaremetalDriver</code>	(class in <i>ironic-client.osc.v1.baremetal_driver</i>), 129
<code>list()</code>	(<i>ironicclient.v1.inspection_rule.InspectionRuleManager</i> method), 182	<code>ListBaremetalDriverProperty</code>	(class in <i>ironicclient.osc.v1.baremetal_driver</i>), 129
<code>list()</code>	(<i>ironicclient.v1.node.NodeManager</i> method), 185	<code>ListBaremetalDriverRaidProperty</code>	(class in <i>ironicclient.osc.v1.baremetal_driver</i>), 130
<code>list()</code>	(<i>ironicclient.v1.port.PortManager</i> method), 197	<code>ListBaremetalInspectionRule</code>	(class in <i>ironic-client.osc.v1.baremetal_inspection_rule</i>), 133
<code>list()</code>	(<i>ironicclient.v1.portgroup.PortgroupManager</i> method), 199	<code>ListBaremetalNode</code>	(class in <i>ironic-client.osc.v1.baremetal_node</i>), 141
<code>list()</code>	(<i>ironicclient.v1.runbook.RunbookManager</i> method), 203	<code>ListBaremetalPort</code>	(class in <i>ironic-client.osc.v1.baremetal_port</i>), 154
<code>list()</code>	(<i>ironicclient.v1.shard.ShardManager</i> method), 205	<code>ListBaremetalPortGroup</code>	(class in <i>ironic-client.osc.v1.baremetal_portgroup</i>), 156
<code>list()</code>	(<i>ironicclient.v1.volume_connector.VolumeConnectorManager</i> method), 206	<code>ListBaremetalRunbook</code>	(class in <i>ironic-client.osc.v1.baremetal_runbook</i>), 159
<code>list()</code>	(<i>ironicclient.v1.volume_target.VolumeTargetManager</i> method), 207	<code>ListBaremetalShard</code>	(class in <i>ironic-client.osc.v1.baremetal_shard</i>), 162
<code>list_bios_settings()</code>	(<i>ironic-client.v1.node.NodeManager</i> method), 187	<code>ListBaremetalVolumeConnector</code>	(class in <i>ironic-client.osc.v1.baremetal_volume_connector</i>), 163
<code>list_children_of_node()</code>	(<i>ironic-client.v1.node.NodeManager</i> method), 187	<code>ListBaremetalVolumeTarget</code>	(class in <i>ironic-client.osc.v1.baremetal_volume_target</i>), 166
<code>list_firmware_components()</code>	(<i>ironic-client.v1.node.NodeManager</i> method), 188	<code>ListBIOSSettingBaremetalNode</code>	(class in <i>ironicclient.osc.v1.baremetal_node</i>), 141
<code>list_nodes()</code>	(<i>ironic-client.v1.chassis.ChassisManager</i> method), 173	<code>ListCommandArgs</code>	(class in <i>ironicclient.common.utils</i>), 115
<code>list_ports()</code>	(<i>ironic-client.v1.node.NodeManager</i> method), 188	<code>ListCommandArgs</code>	(class in <i>ironicclient.osc.command</i>), 169
<code>list_ports()</code>	(<i>ironic-client.v1.portgroup.PortgroupManager</i> method), 200	<code>ListFirmwareComponentBaremetalNode</code>	(class in <i>ironic-client.osc.v1.baremetal_node</i>), 142
<code>list_volume_connectors()</code>	(<i>ironic-client.v1.node.NodeManager</i> method), 189		

`ListTraitsBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 142

`ListTraitsBaremetalRunbook` (class in `ironic-client.osc.v1.baremetal_runbook`), 160

`load_commands()` (`ironic-client.shell.CommandManager` method), 210

`load_from_file()` (in module `ironic-client.v1.create_resources`), 179

`log (ironicclient.osc.command.Command attribute)`, 169

`log (ironicclient.osc.command.Lister attribute)`, 169

`log (ironicclient.osc.command.ShowOne attribute)`, 169

`log (ironicclient.osc.v1.baremetal_allocation.CreateBaremetalAllocation attribute)`, 119

`log (ironicclient.osc.v1.baremetal_allocation.DeleteBaremetalAllocation attribute)`, 120

`log (ironicclient.osc.v1.baremetal_allocation.ListBaremetalAllocation attribute)`, 120

`log (ironicclient.osc.v1.baremetal_allocation.SetBaremetalAllocation attribute)`, 121

`log (ironicclient.osc.v1.baremetal_allocation.ShowBaremetalAllocation attribute)`, 121

`log (ironicclient.osc.v1.baremetal_allocation.UnsetBaremetalAllocation attribute)`, 122

`log (ironicclient.osc.v1.baremetal_chassis.CreateBaremetalChassis attribute)`, 122

`log (ironicclient.osc.v1.baremetal_chassis.DeleteBaremetalChassis attribute)`, 123

`log (ironicclient.osc.v1.baremetal_chassis.ListBaremetalChassis attribute)`, 123

`log (ironicclient.osc.v1.baremetal_chassis.SetBaremetalChassis attribute)`, 123

`log (ironicclient.osc.v1.baremetal_chassis.ShowBaremetalChassis attribute)`, 124

`log (ironicclient.osc.v1.baremetal_chassis.UnsetBaremetalChassis attribute)`, 124

`log (ironicclient.osc.v1.baremetal_conductor.ListBaremetalConductor attribute)`, 125

`log (ironicclient.osc.v1.baremetal_conductor.ShowBaremetalConductor attribute)`, 125

`log (ironicclient.osc.v1.baremetal_create.CreateBaremetal attribute)`, 126

`log (ironicclient.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplate attribute)`, 126

`log (ironicclient.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate attribute)`, 127

`log (ironicclient.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplate attribute)`, 127

`log (ironicclient.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate attribute)`, 128

`log (ironicclient.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate attribute)`, 128

`log (ironicclient.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate attribute)`, 129

`log (ironicclient.osc.v1.baremetal_driver.ListBaremetalDriver attribute)`, 129

`log (ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverProperties attribute)`, 130

`log (ironicclient.osc.v1.baremetal_driver.ListBaremetalDriverRaids attribute)`, 130

`log (ironicclient.osc.v1.baremetal_driver.PassthruCallBaremetalDriver attribute)`, 131

`log (ironicclient.osc.v1.baremetal_driver.PassthruListBaremetalDriver attribute)`, 131

`log (ironicclient.osc.v1.baremetal_driver.ShowBaremetalDriver attribute)`, 131

`log (ironicclient.osc.v1.baremetal_inspection_rule.CreateBaremetalInspectionRule attribute)`, 132

`log (ironicclient.osc.v1.baremetal_inspection_rule.DeleteBaremetalInspectionRule attribute)`, 133

`log (ironicclient.osc.v1.baremetal_inspection_rule.ListBaremetalInspectionRule attribute)`, 133

`log (ironicclient.osc.v1.baremetal_inspection_rule.SetBaremetalInspectionRule attribute)`, 134

`log (ironicclient.osc.v1.baremetal_inspection_rule.ShowBaremetalInspectionRule attribute)`, 134

`log (ironicclient.osc.v1.baremetal_inspection_rule.UnsetBaremetalInspectionRule attribute)`, 135

`log (ironicclient.osc.v1.baremetal_node.AbortBaremetalNode attribute)`, 135

`log (ironicclient.osc.v1.baremetal_node.AddTraitBaremetalNode attribute)`, 136

`log (ironicclient.osc.v1.baremetal_node.AdoptBaremetalNode attribute)`, 136

`log (ironicclient.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode attribute)`, 137

`log (ironicclient.osc.v1.baremetal_node.BootdeviceSetBaremetalNode attribute)`, 137

`log (ironicclient.osc.v1.baremetal_node.BootdeviceShowBaremetalNode attribute)`, 137

`log (ironicclient.osc.v1.baremetal_node.BootmodeSetBaremetalNode attribute)`, 138

`log (ironicclient.osc.v1.baremetal_node.CleanBaremetalNode attribute)`, 138

`log (ironicclient.osc.v1.baremetal_node.ConsoleDisableBaremetalNode attribute)`, 138

`log (ironicclient.osc.v1.baremetal_node.ConsoleEnableBaremetalNode attribute)`, 138

`log (ironicclient.osc.v1.baremetal_node.ConsoleShowBaremetalNode attribute)`, 138

`attribute`), 139

`log (ironicclient.osc.v1.baremetal_node.CreateBaremetalNode`
`attribute`), 139

`log (ironicclient.osc.v1.baremetal_node.DeleteBaremetalNode`
`attribute`), 140

`log (ironicclient.osc.v1.baremetal_node.DeployBaremetalNode`
`attribute`), 140

`log (ironicclient.osc.v1.baremetal_node.InjectNmiBaremetalNode`
`attribute`), 140

`log (ironicclient.osc.v1.baremetal_node.InspectBaremetalNode`
`attribute`), 141

`log (ironicclient.osc.v1.baremetal_node.ListBaremetalNode`
`attribute`), 142

`log (ironicclient.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode`
`attribute`), 141

`log (ironicclient.osc.v1.baremetal_node.ListFirmwareComponentsBaremetalNode`
`attribute`), 142

`log (ironicclient.osc.v1.baremetal_node.ListTraitsBaremetalNode`
`attribute`), 142

`log (ironicclient.osc.v1.baremetal_node.MaintenanceSetBaremetalNode`
`attribute`), 143

`log (ironicclient.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode`
`attribute`), 143

`log (ironicclient.osc.v1.baremetal_node.ManageBaremetalNode`
`attribute`), 144

`log (ironicclient.osc.v1.baremetal_node.NodeChildrenList`
`attribute`), 144

`log (ironicclient.osc.v1.baremetal_node.NodeHistoryEventGet`
`attribute`), 144

`log (ironicclient.osc.v1.baremetal_node.NodeHistoryList`
`attribute`), 145

`log (ironicclient.osc.v1.baremetal_node.NodeInventorySave`
`attribute`), 145

`log (ironicclient.osc.v1.baremetal_node.PassthruCallBaremetalNode`
`attribute`), 145

`log (ironicclient.osc.v1.baremetal_node.PassthruListBaremetalNode`
`attribute`), 146

`log (ironicclient.osc.v1.baremetal_node.PowerBaremetalNode`
`attribute`), 146

`log (ironicclient.osc.v1.baremetal_node.PowerOffBaremetalNode`
`attribute`), 146

`log (ironicclient.osc.v1.baremetal_node.PowerOnBaremetalNode`
`attribute`), 147

`log (ironicclient.osc.v1.baremetal_node.ProvideBaremetalNode`
`attribute`), 147

`log (ironicclient.osc.v1.baremetal_node.ProvisionStateBaremetalNode`
`attribute`), 147

`log (ironicclient.osc.v1.baremetal_node.ProvisionStateWithWait`
`attribute`), 148

`log (ironicclient.osc.v1.baremetal_node.RebootBaremetalNode`
`attribute`), 148

`log (ironicclient.osc.v1.baremetal_node.RebuildBaremetalNode`
`attribute`), 148

`log (ironicclient.osc.v1.baremetal_node.RemoveTraitBaremetalNode`
`attribute`), 149

`log (ironicclient.osc.v1.baremetal_node.RescueBaremetalNode`
`attribute`), 149

`log (ironicclient.osc.v1.baremetal_node.SecurebootOffBaremetalNode`
`attribute`), 149

`log (ironicclient.osc.v1.baremetal_node.SecurebootOnBaremetalNode`
`attribute`), 150

`log (ironicclient.osc.v1.baremetal_node.ServiceBaremetalNode`
`attribute`), 150

`log (ironicclient.osc.v1.baremetal_node.SetBaremetalNode`
`attribute`), 150

`log (ironicclient.osc.v1.baremetal_node.ShowBaremetalNode`
`attribute`), 151

`log (ironicclient.osc.v1.baremetal_node.UndeployBaremetalNode`
`attribute`), 151

`log (ironicclient.osc.v1.baremetal_node.UnholdBaremetalNode`
`attribute`), 151

`log (ironicclient.osc.v1.baremetal_node.UnrescueBaremetalNode`
`attribute`), 152

`log (ironicclient.osc.v1.baremetal_node.UnsetBaremetalNode`
`attribute`), 152

`log (ironicclient.osc.v1.baremetal_node.ValidateBaremetalNode`
`attribute`), 152

`log (ironicclient.osc.v1.baremetal_node.VifAttachBaremetalNode`
`attribute`), 153

`log (ironicclient.osc.v1.baremetal_node.VifDetachBaremetalNode`
`attribute`), 153

`log (ironicclient.osc.v1.baremetal_node.VifListBaremetalNode`
`attribute`), 153

`log (ironicclient.osc.v1.baremetal_port.CreateBaremetalPort`
`attribute`), 154

`log (ironicclient.osc.v1.baremetal_port.DeleteBaremetalPort`
`attribute`), 154

`log (ironicclient.osc.v1.baremetal_port.ListBaremetalPort`
`attribute`), 154

`log (ironicclient.osc.v1.baremetal_port.SetBaremetalPort`
`attribute`), 155

`log (ironicclient.osc.v1.baremetal_port.ShowBaremetalPort`
`attribute`), 155

`log (ironicclient.osc.v1.baremetal_port.UnsetBaremetalPort`
`attribute`), 155

`log (ironicclient.osc.v1.baremetal_portgroup.CreateBaremetalPort`
`attribute`), 156

`log (ironicclient.osc.v1.baremetal_portgroup.DeleteBaremetalPort`
`attribute`), 156

`log (ironicclient.osc.v1.baremetal_portgroup.ListBaremetalPort`
`attribute`), 156

`log (ironicclient.osc.v1.baremetal_portgroup.SetBaremetalPort`
`attribute`), 157

attribute), 157	M
log (ironicclient.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup attribute), 158	main() (in module ironicclient.shell), 210
log (ironicclient.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup attribute), 158	MaintenanceSetBaremetalNode (class in iron- client.osc.v1.baremetal_node), 143
log (ironicclient.osc.v1.baremetal_runbook.AddTraitBaremetalRunbook attribute), 158	MaintenanceUnsetBaremetalNode (class in ironicclient.osc.v1.baremetal_node), 143
log (ironicclient.osc.v1.baremetal_runbook.CreateBaremetalRunbook attribute), 159	make_client() (in module ironic- client.osc.plugin), 170
log (ironicclient.osc.v1.baremetal_runbook.DeleteBaremetalRunbook attribute), 159	make_configdrive() (in module ironic- client.common.utils), 118
log (ironicclient.osc.v1.baremetal_runbook.ListBaremetalRunbook attribute), 160	ManageBaremetalNode (class in ironic- client.osc.v1.baremetal_node), 143
log (ironicclient.osc.v1.baremetal_runbook.ListTraitsBaremetalRunbook attribute), 160	Manager (class in ironicclient.common.base), 113
log (ironicclient.osc.v1.baremetal_runbook.RemoveTraitBaremetalRunbook attribute), 160	ManagerProtocol (class in ironic- client.common.apiclient.base), 93
log (ironicclient.osc.v1.baremetal_runbook.SetBaremetalRunbook attribute), 161	marker (in module ironicclient.common.utils.ListCommandArgs attribute), 115
log (ironicclient.osc.v1.baremetal_runbook.ShowBaremetalRunbook attribute), 161	message (ironic- client.common.apiclient.exceptions.BadGateway attribute), 94
log (ironicclient.osc.v1.baremetal_runbook.UnsetBaremetalRunbook attribute), 162	message (ironic- client.common.apiclient.exceptions.BadRequest attribute), 95
log (ironicclient.osc.v1.baremetal_shard.ListBaremetalShard attribute), 162	message (ironic- client.common.apiclient.exceptions.Conflict attribute), 95
log (ironicclient.osc.v1.baremetal_volume_connector.CreateBaremetalVolumeConnector attribute), 162	message (ironic- client.common.apiclient.exceptions.ExpectationFailed attribute), 96
log (ironicclient.osc.v1.baremetal_volume_connector.DeleteBaremetalVolumeConnector attribute), 163	message (ironic- client.common.apiclient.exceptions.Forbidden attribute), 96
log (ironicclient.osc.v1.baremetal_volume_connector.ListBaremetalVolumeConnector attribute), 163	message (ironic- client.common.apiclient.exceptions.GatewayTimeout attribute), 97
log (ironicclient.osc.v1.baremetal_volume_connector.SetBaremetalVolumeConnector attribute), 164	message (ironic- client.common.apiclient.exceptions.Gone attribute), 97
log (ironicclient.osc.v1.baremetal_volume_connector.ShowBaremetalVolumeConnector attribute), 165	message (ironic- client.common.apiclient.exceptions.HTTPClientError attribute), 98
log (ironicclient.osc.v1.baremetal_volume_connector.UnsetBaremetalVolumeConnector attribute), 165	message (ironic- client.common.apiclient.exceptions.HttpError attribute), 98
log (ironicclient.osc.v1.baremetal_volume_target.CreateBaremetalVolumeTarget attribute), 166	message (ironic- client.common.apiclient.exceptions.HttpNotImplemented attribute), 99
log (ironicclient.osc.v1.baremetal_volume_target.DeleteBaremetalVolumeTarget attribute), 166	message (ironic- client.common.apiclient.exceptions.HTTPRedirection attribute), 98
log (ironicclient.osc.v1.baremetal_volume_target.ListBaremetalVolumeTarget attribute), 167	message (ironic-
log (ironicclient.osc.v1.baremetal_volume_target.SetBaremetalVolumeTarget attribute), 167	
log (ironicclient.osc.v1.baremetal_volume_target.ShowBaremetalVolumeTarget attribute), 168	
log (ironicclient.osc.v1.baremetal_volume_target.UnsetBaremetalVolumeTarget attribute), 168	

`client.common.apiclient.exceptions.HttpServerError` `client.common.apiclient.exceptions.UnprocessableEntity`
 attribute), 100 attribute), 111

message (ironic- message (ironic-
`client.common.apiclient.exceptions.HttpVersionNotSupported` `client.common.apiclient.exceptions.UnsupportedMediaType`
 attribute), 101 attribute), 112

message (ironic- method
`client.common.apiclient.exceptions.InternalServerError` `openstack-baremetal-driver-passthru-call`
 attribute), 102 command line option, 25

message (ironic- `openstack-baremetal-node-passthru-call`
`client.common.apiclient.exceptions.LengthRequired` command line option, 48
 attribute), 102 `MethodNotAllowed`, 103

message (ironic- module
`client.common.apiclient.exceptions.MethodNotAllowed` `ironicclient`, 210
 attribute), 103 `ironicclient.client`, 208

message (ironic- `ironicclient.common`, 119
`client.common.apiclient.exceptions.MultipleChoices` `ironicclient.common.apiclient`, 113
 attribute), 103 `ironicclient.common.apiclient.base`,
 93

message (ironic- 93
`client.common.apiclient.exceptions.NotAcceptable` `ironicclient.common.apiclient.exceptions`,
 attribute), 104 93

message (ironic- `ironicclient.common.base`, 113
`client.common.apiclient.exceptions.NotFound` `ironicclient.common.filecache`, 114
 attribute), 104 `ironicclient.common.http`, 114

message (ironic- `ironicclient.common.i18n`, 115
`client.common.apiclient.exceptions.PaymentRequired` `ironicclient.common.utils`, 115
 attribute), 105 `ironicclient.exc`, 209

message (ironic- `ironicclient.osc`, 170
`client.common.apiclient.exceptions.PreconditionFailed` `ironicclient.osc.command`, 169
 attribute), 106 `ironicclient.osc.plugin`, 169

message (ironic- `ironicclient.osc.v1`, 169
`client.common.apiclient.exceptions.ProxyAuthenticationRequired` `ironicclient.osc.v1.baremetal_allocation`,
 attribute), 107 119

message (ironic- `ironicclient.osc.v1.baremetal_chassis`,
`client.common.apiclient.exceptions.RequestedRangeNotSatisfiable` 125
 attribute), 110 `ironicclient.osc.v1.baremetal_conductor`,
 124

message (ironic- 124
`client.common.apiclient.exceptions.RequestEntityTooLarge` `ironicclient.osc.v1.baremetal_create`,
 attribute), 107 125

message (ironic- `ironicclient.osc.v1.baremetal_deploy_template`,
`client.common.apiclient.exceptions.RequestTimeout` 126
 attribute), 107 `ironicclient.osc.v1.baremetal_driver`,
 129

message (ironic- 129
`client.common.apiclient.exceptions.RequestUriTooBig` `ironicclient.osc.v1.baremetal_inspection_rule`,
 attribute), 108 132

message (ironic- `ironicclient.osc.v1.baremetal_node`,
`client.common.apiclient.exceptions.ServiceUnavailable` 135
 attribute), 110 `ironicclient.osc.v1.baremetal_port`,
 154

message (ironic- 154
`client.common.apiclient.exceptions.Unauthorized` `ironicclient.osc.v1.baremetal_portgroup`,
 attribute), 111 156

message (ironic- `ironicclient.osc.v1.baremetal_runbook`,

- 158
- ironicclient.osc.v1.baremetal_shard, 162
- ironicclient.osc.v1.baremetal_volume_connector, 162
- ironicclient.osc.v1.baremetal_volume_target, 165
- ironicclient.shell, 210
- ironicclient.v1, 208
- ironicclient.v1.allocation, 170
- ironicclient.v1.chassis, 172
- ironicclient.v1.client, 174
- ironicclient.v1.conductor, 175
- ironicclient.v1.create_resources, 176
- ironicclient.v1.deploy_template, 179
- ironicclient.v1.driver, 180
- ironicclient.v1.events, 182
- ironicclient.v1.inspection_rule, 182
- ironicclient.v1.node, 183
- ironicclient.v1.port, 197
- ironicclient.v1.portgroup, 198
- ironicclient.v1.resource_fields, 201
- ironicclient.v1.runbook, 203
- ironicclient.v1.shard, 205
- ironicclient.v1.utils, 205
- ironicclient.v1.volume_connector, 205
- ironicclient.v1.volume_target, 207
- MultipleChoices, 103
- N
- name
 - openstack-baremetal-deploy-template-create
 - command line option, 20
- NAME_ATTR (ironic-client.common.apiclient.base.Resource attribute), 93
- negotiate_api_version() (ironic-client.v1.client.Client method), 175
- negotiate_version() (ironic-client.common.http.VersionNegotiationMixin method), 114
- node
 - openstack-baremetal-node-abort
 - command line option, 29
 - openstack-baremetal-node-add-trait
 - command line option, 29
 - openstack-baremetal-node-adopt
 - command line option, 30
 - openstack-baremetal-node-bios-setting-list
 - command line option, 31
 - openstack-baremetal-node-bios-setting-show
 - command line option, 32
 - openstack-baremetal-node-boot-device-set
 - command line option, 32
 - openstack-baremetal-node-boot-device-show
 - command line option, 33
 - openstack-baremetal-node-boot-mode-set
 - command line option, 33
 - openstack-baremetal-node-children-list
 - command line option, 34
 - openstack-baremetal-node-clean
 - command line option, 35
 - openstack-baremetal-node-console-disable
 - command line option, 35
 - openstack-baremetal-node-console-enable
 - command line option, 35
 - openstack-baremetal-node-console-show
 - command line option, 36
 - openstack-baremetal-node-delete
 - command line option, 39
 - openstack-baremetal-node-deploy
 - command line option, 40
 - openstack-baremetal-node-firmware-list
 - command line option, 41
 - openstack-baremetal-node-history-get
 - command line option, 42
 - openstack-baremetal-node-history-list
 - command line option, 43
 - openstack-baremetal-node-inject-nmi
 - command line option, 43
 - openstack-baremetal-node-inspect
 - command line option, 43
 - openstack-baremetal-node-inventory-save
 - command line option, 44
 - openstack-baremetal-node-maintenance-set
 - command line option, 47
 - openstack-baremetal-node-maintenance-unset
 - command line option, 47
 - openstack-baremetal-node-manage
 - command line option, 47
 - openstack-baremetal-node-passthru-call
 - command line option, 48
 - openstack-baremetal-node-passthru-list
 - command line option, 49
 - openstack-baremetal-node-power-off
 - command line option, 49
 - openstack-baremetal-node-power-on
 - command line option, 50
 - openstack-baremetal-node-provide
 - command line option, 50

openstack-baremetal-node-reboot	NotFound, 104
command line option, 50	NoUniqueMatch, 103
openstack-baremetal-node-rebuild	
command line option, 51	O
openstack-baremetal-node-remove-trait	openstack-baremetal-allocation-create
command line option, 51	command line option
openstack-baremetal-node-rescue	--candidate-node, 8
command line option, 52	--column, 8
openstack-baremetal-node-secure-boot-off	--extra, 8
command line option, 52	--fit-width, 8
openstack-baremetal-node-secure-boot-on	--format, 8
command line option, 52	--max-width, 8
openstack-baremetal-node-service	--name, 8
command line option, 53	--node, 9
openstack-baremetal-node-set	--noindent, 8
command line option, 57	--owner, 8
openstack-baremetal-node-show	--prefix, 8
command line option, 58	--print-empty, 8
openstack-baremetal-node-trait-add	--resource-class, 8
command line option, 58	--trait, 8
openstack-baremetal-node-trait-list	--uuid, 8
command line option, 59	--wait, 8
openstack-baremetal-node-trait-remove	-c, 8
command line option, 59	-f, 8
openstack-baremetal-node-undeploy	openstack-baremetal-allocation-delete
command line option, 60	command line option
openstack-baremetal-node-unhold	allocation, 9
command line option, 60	openstack-baremetal-allocation-list
openstack-baremetal-node-unrescue	command line option
command line option, 60	--column, 9
openstack-baremetal-node-unset	--fields, 10
command line option, 63	--fit-width, 10
openstack-baremetal-node-validate	--format, 9
command line option, 64	--limit, 10
openstack-baremetal-node-vif-attach	--long, 10
command line option, 65	--marker, 10
openstack-baremetal-node-vif-detach	--max-width, 10
command line option, 65	--node, 10
openstack-baremetal-node-vif-list	--noindent, 9
command line option, 66	--owner, 10
Node (class in <i>ironicclient.v1.node</i>), 183	--print-empty, 10
NodeChildrenList (class in <i>ironic-</i>	--quote, 9
<i>client.osc.v1.baremetal_node</i>), 144	--resource-class, 10
NodeHistoryEventGet (class in <i>ironic-</i>	--sort, 10
<i>client.osc.v1.baremetal_node</i>), 144	--sort-ascending, 10
NodeHistoryList (class in <i>ironic-</i>	--sort-column, 10
<i>client.osc.v1.baremetal_node</i>), 144	--sort-descending, 10
NodeInventorySave (class in <i>ironic-</i>	--state, 10
<i>client.osc.v1.baremetal_node</i>), 145	-c, 9
NodeManager (class in <i>ironicclient.v1.node</i>), 183	-f, 9
NotAcceptable, 104	

```
openstack-baremetal-allocation-set
    command line option
    --extra, 11
    --name, 11
    allocation, 11
openstack-baremetal-allocation-show
    command line option
    --column, 11
    --fields, 12
    --fit-width, 11
    --format, 11
    --max-width, 11
    --noindent, 11
    --prefix, 11
    --print-empty, 11
    -c, 11
    -f, 11
    id, 12
openstack-baremetal-allocation-unset
    command line option
    --extra, 12
    --name, 12
    allocation, 12
openstack-baremetal-chassis-create
    command line option
    --column, 12
    --description, 13
    --extra, 13
    --fit-width, 13
    --format, 12
    --max-width, 13
    --noindent, 12
    --prefix, 13
    --print-empty, 13
    --uuid, 13
    -c, 12
    -f, 12
openstack-baremetal-chassis-delete
    command line option
    chassis, 13
openstack-baremetal-chassis-list
    command line option
    --column, 14
    --fields, 14
    --fit-width, 14
    --format, 14
    --limit, 14
    --long, 14
    --marker, 14
    --max-width, 14
    --noindent, 14
    --print-empty, 14
    --quote, 14
    --sort, 14
    --sort-ascending, 14
    --sort-column, 14
    --sort-descending, 14
    -c, 14
    -f, 14
openstack-baremetal-chassis-set
    command line option
    --description, 15
    --extra, 15
    chassis, 15
openstack-baremetal-chassis-show
    command line option
    --column, 15
    --fields, 16
    --fit-width, 16
    --format, 15
    --max-width, 15
    --noindent, 15
    --prefix, 15
    --print-empty, 16
    -c, 15
    -f, 15
    chassis, 16
openstack-baremetal-chassis-unset
    command line option
    --description, 16
    --extra, 16
    chassis, 16
openstack-baremetal-conductor-list
    command line option
    --column, 17
    --fields, 17
    --fit-width, 17
    --format, 17
    --limit, 17
    --long, 17
    --marker, 17
    --max-width, 17
    --noindent, 17
    --print-empty, 17
    --quote, 17
    --sort, 17
    --sort-ascending, 17
    --sort-column, 17
    --sort-descending, 17
    -c, 17
    -f, 17
openstack-baremetal-conductor-show
```

- command line option
- column, 18
- fields, 18
- fit-width, 18
- format, 18
- max-width, 18
- noindent, 18
- prefix, 18
- print-empty, 18
- c, 18
- f, 18
- conductor, 18
- openstack-baremetal-create command
 - line option
 - file, 19
- openstack-baremetal-deploy-template-create
 - command line option
 - column, 19
 - extra, 20
 - fit-width, 19
 - format, 19
 - max-width, 19
 - noindent, 19
 - prefix, 19
 - print-empty, 19
 - steps, 20
 - uuid, 20
 - c, 19
 - f, 19
 - name, 20
- openstack-baremetal-deploy-template-delete
 - command line option
 - template, 20
- openstack-baremetal-deploy-template-list
 - command line option
 - column, 20
 - fields, 21
 - fit-width, 21
 - format, 20
 - limit, 21
 - long, 21
 - marker, 21
 - max-width, 21
 - noindent, 21
 - print-empty, 21
 - quote, 20
 - sort, 21
 - sort-ascending, 21
 - sort-column, 21
 - sort-descending, 21
 - c, 20
 - f, 20
- openstack-baremetal-deploy-template-set
 - command line option
 - extra, 22
 - name, 22
 - steps, 22
 - template, 22
- openstack-baremetal-deploy-template-show
 - command line option
 - column, 22
 - fields, 23
 - fit-width, 22
 - format, 22
 - max-width, 22
 - noindent, 22
 - prefix, 22
 - print-empty, 22
 - c, 22
 - f, 22
 - template, 23
- openstack-baremetal-deploy-template-unset
 - command line option
 - extra, 23
 - template, 23
- openstack-baremetal-driver-list
 - command line option
 - column, 23
 - fields, 24
 - fit-width, 24
 - format, 23
 - long, 24
 - max-width, 24
 - noindent, 23
 - print-empty, 24
 - quote, 23
 - sort-ascending, 24
 - sort-column, 24
 - sort-descending, 24
 - type, 24
 - c, 23
 - f, 23
- openstack-baremetal-driver-passthru-call
 - command line option
 - arg, 25
 - column, 25
 - fit-width, 25
 - format, 24
 - http-method, 25
 - max-width, 25
 - noindent, 25
 - prefix, 25

```

--print-empty, 25
-c, 25
-f, 24
driver, 25
method, 25
openstack-baremetal-driver-passthru-list
    command line option
--column, 26
--fit-width, 26
--format, 25
--max-width, 26
--noindent, 26
--print-empty, 26
--quote, 26
--sort-ascending, 26
--sort-column, 26
--sort-descending, 26
-c, 26
-f, 25
driver, 26
openstack-baremetal-driver-property-list
    command line option
--column, 27
--fit-width, 27
--format, 26
--max-width, 27
--noindent, 27
--print-empty, 27
--quote, 27
--sort-ascending, 27
--sort-column, 27
--sort-descending, 27
-c, 27
-f, 26
driver, 27
openstack-baremetal-driver-raid-property-list
    command line option
--column, 28
--fit-width, 28
--format, 27
--max-width, 28
--noindent, 28
--print-empty, 28
--quote, 28
--sort-ascending, 28
--sort-column, 28
--sort-descending, 28
-c, 28
-f, 27
driver, 28
openstack-baremetal-driver-show
    command line option
--column, 29
--fields, 29
--fit-width, 29
--format, 28
--max-width, 29
--noindent, 29
--prefix, 29
--print-empty, 29
-c, 29
-f, 28
driver, 29
openstack-baremetal-node-abort
    command
    line option
    node, 29
openstack-baremetal-node-add-trait
    command line option
    node, 29
    trait, 29
openstack-baremetal-node-adopt
    command
    line option
--wait, 30
    node, 30
openstack-baremetal-node-bios-setting-list
    command line option
--column, 30
--fields, 31
--fit-width, 30
--format, 30
--long, 31
--max-width, 30
--noindent, 30
--print-empty, 30
--quote, 30
--sort-ascending, 31
--sort-column, 31
--sort-descending, 31
-c, 30
-f, 30
    node, 31
openstack-baremetal-node-bios-setting-show
    command line option
--column, 31
--fit-width, 32
--format, 31
--max-width, 31
--noindent, 31
--prefix, 31
--print-empty, 32
-c, 31
-f, 31

```

- node, [32](#)
- setting, [32](#)
- openstack-baremetal-node-boot-device-set
 - command line option
- persistent, [32](#)
- device, [32](#)
- node, [32](#)
- openstack-baremetal-node-boot-device-show
 - command line option
- column, [33](#)
- fit-width, [33](#)
- format, [32](#)
- max-width, [33](#)
- noindent, [33](#)
- prefix, [33](#)
- print-empty, [33](#)
- supported, [33](#)
- c, [33](#)
- f, [32](#)
- node, [33](#)
- openstack-baremetal-node-boot-mode-set
 - command line option
- boot_mode, [33](#)
- node, [33](#)
- openstack-baremetal-node-children-list
 - command line option
- column, [34](#)
- fit-width, [34](#)
- format, [34](#)
- max-width, [34](#)
- noindent, [34](#)
- prefix, [34](#)
- print-empty, [34](#)
- c, [34](#)
- f, [34](#)
- node, [34](#)
- openstack-baremetal-node-clean
 - command line option
- clean-steps, [34](#)
- disable-ramdisk, [35](#)
- runbook, [34](#)
- wait, [34](#)
- node, [35](#)
- openstack-baremetal-node-console-disable
 - command line option
- node, [35](#)
- openstack-baremetal-node-console-enable
 - command line option
- node, [35](#)
- openstack-baremetal-node-console-show
 - command line option
- column, [35](#)
- fit-width, [36](#)
- format, [35](#)
- max-width, [36](#)
- noindent, [35](#)
- prefix, [36](#)
- print-empty, [36](#)
- c, [35](#)
- f, [35](#)
- node, [36](#)
- openstack-baremetal-node-create
 - command line option
- automated-clean, [39](#)
- bios-interface, [38](#)
- boot-interface, [38](#)
- chassis-uuid, [37](#)
- column, [37](#)
- conductor-group, [39](#)
- console-interface, [38](#)
- deploy-interface, [38](#)
- description, [39](#)
- disable-power-off, [39](#)
- driver, [37](#)
- driver-info, [37](#)
- extra, [37](#)
- firmware-interface, [39](#)
- fit-width, [37](#)
- format, [37](#)
- inspect-interface, [38](#)
- instance-name, [38](#)
- lessee, [39](#)
- management-interface, [38](#)
- max-width, [37](#)
- name, [38](#)
- network-data, [38](#)
- network-interface, [38](#)
- no-automated-clean, [39](#)
- noindent, [37](#)
- owner, [39](#)
- parent-node, [39](#)
- power-interface, [38](#)
- prefix, [37](#)
- print-empty, [37](#)
- property, [37](#)
- raid-interface, [38](#)
- rescue-interface, [38](#)
- resource-class, [39](#)
- shard, [39](#)
- storage-interface, [38](#)
- uuid, [37](#)
- vendor-interface, [39](#)

- c, 37
- f, 37
- openstack-baremetal-node-delete
 - command line option
 - node, 39
- openstack-baremetal-node-deploy
 - command line option
 - config-drive, 40
 - deploy-steps, 40
 - wait, 40
 - node, 40
- openstack-baremetal-node-firmware-list
 - command line option
 - column, 40
 - fit-width, 41
 - format, 40
 - max-width, 41
 - noindent, 41
 - print-empty, 41
 - quote, 41
 - sort-ascending, 41
 - sort-column, 41
 - sort-descending, 41
 - c, 40
 - f, 40
 - node, 41
- openstack-baremetal-node-history-get
 - command line option
 - column, 41
 - fit-width, 42
 - format, 41
 - max-width, 42
 - noindent, 42
 - prefix, 42
 - print-empty, 42
 - c, 41
 - f, 41
 - event, 42
 - node, 42
- openstack-baremetal-node-history-list
 - command line option
 - column, 42
 - fit-width, 43
 - format, 42
 - long, 43
 - max-width, 43
 - noindent, 42
 - print-empty, 43
 - quote, 42
 - sort-ascending, 43
 - sort-column, 43
 - sort-descending, 43
 - c, 42
 - f, 42
 - node, 43
- openstack-baremetal-node-inject-nmi
 - command line option
 - node, 43
- openstack-baremetal-node-inspect
 - command line option
 - wait, 43
 - node, 43
- openstack-baremetal-node-inventory-save
 - command line option
 - file, 44
 - node, 44
- openstack-baremetal-node-list
 - command line option
 - associated, 46
 - chassis, 46
 - column, 45
 - conductor, 46
 - conductor-group, 46
 - description-contains, 46
 - driver, 46
 - fault, 46
 - fields, 46
 - fit-width, 45
 - format, 45
 - include-children, 47
 - instance-name, 46
 - lessee, 46
 - limit, 45
 - long, 46
 - maintenance, 45
 - marker, 45
 - max-width, 45
 - no-maintenance, 45
 - no-retired, 46
 - noindent, 45
 - owner, 46
 - parent-node, 47
 - print-empty, 45
 - provision-state, 46
 - quote, 45
 - resource-class, 46
 - retired, 45
 - sharded, 46
 - shards, 46
 - sort, 45
 - sort-ascending, 45
 - sort-column, 45

- `--sort-descending`, 45
 - `--unassociated`, 46
 - `--unsharded`, 46
 - `-c`, 45
 - `-f`, 45
- `openstack-baremetal-node-maintenance-set`
 - command line option
 - `--reason`, 47
 - node, 47
- `openstack-baremetal-node-maintenance-unset`
 - command line option
 - node, 51
- `openstack-baremetal-node-manage`
 - command line option
 - `--wait`, 47
 - node, 47
- `openstack-baremetal-node-passthru-call`
 - command line option
 - `--arg`, 48
 - `--http-method`, 48
 - method, 48
 - node, 48
- `openstack-baremetal-node-passthru-list`
 - command line option
 - `--column`, 48
 - `--fit-width`, 49
 - `--format`, 48
 - `--max-width`, 49
 - `--noindent`, 48
 - `--print-empty`, 49
 - `--quote`, 48
 - `--sort-ascending`, 49
 - `--sort-column`, 49
 - `--sort-descending`, 49
 - `-c`, 48
 - `-f`, 48
 - node, 49
- `openstack-baremetal-node-power-off`
 - command line option
 - `--power-timeout`, 49
 - `--soft`, 49
 - node, 49
- `openstack-baremetal-node-power-on`
 - command line option
 - `--power-timeout`, 50
 - node, 50
- `openstack-baremetal-node-provide`
 - command line option
 - `--wait`, 50
 - node, 50
- `openstack-baremetal-node-reboot`
 - command line option
 - `--power-timeout`, 50
 - `--soft`, 50
 - node, 50
- `openstack-baremetal-node-rebuild`
 - command line option
 - `--config-drive`, 51
 - `--deploy-steps`, 51
 - `--wait`, 51
 - node, 51
- `openstack-baremetal-node-remove-trait`
 - command line option
 - `--all`, 51
 - node, 51
 - trait, 51
- `openstack-baremetal-node-rescue`
 - command line option
 - `--rescue-password`, 52
 - `--wait`, 52
 - node, 52
- `openstack-baremetal-node-secure-boot-off`
 - command line option
 - node, 52
- `openstack-baremetal-node-secure-boot-on`
 - command line option
 - node, 52
- `openstack-baremetal-node-service`
 - command line option
 - `--disable-ramdisk`, 53
 - `--runbook`, 53
 - `--service-steps`, 53
 - `--wait`, 53
 - node, 53
- `openstack-baremetal-node-set` command line option
 - `--automated-clean`, 56
 - `--bios-interface`, 54
 - `--boot-interface`, 54
 - `--chassis-uuid`, 54
 - `--conductor-group`, 56
 - `--console-interface`, 54
 - `--deploy-interface`, 54
 - `--description`, 57
 - `--disable-power-off`, 57
 - `--driver`, 54
 - `--driver-info`, 56
 - `--enable-power-off`, 57
 - `--extra`, 56
 - `--firmware-interface`, 55
 - `--inspect-interface`, 55
 - `--instance-info`, 56

- instance-name, 54
- instance-uuid, 54
- lessee, 56
- management-interface, 55
- name, 54
- network-data, 55
- network-interface, 55
- no-automated-clean, 56
- owner, 56
- parent-node, 57
- power-interface, 55
- property, 56
- protected, 56
- protected-reason, 56
- raid-interface, 55
- rescue-interface, 55
- reset-bios-interface, 54
- reset-boot-interface, 54
- reset-console-interface, 54
- reset-deploy-interface, 55
- reset-firmware-interface, 55
- reset-inspect-interface, 55
- reset-interfaces, 56
- reset-management-interface, 55
- reset-network-interface, 55
- reset-power-interface, 55
- reset-raid-interface, 55
- reset-rescue-interface, 55
- reset-storage-interface, 55
- reset-vendor-interface, 56
- resource-class, 56
- retired, 56
- retired-reason, 56
- shard, 57
- storage-interface, 55
- target-raid-config, 56
- vendor-interface, 56
- node, 57
- openstack-baremetal-node-show command
 - line option
 - column, 57
 - fields, 58
 - fit-width, 57
 - format, 57
 - instance, 58
 - max-width, 57
 - noindent, 57
 - prefix, 57
 - print-empty, 58
 - c, 57
 - f, 57
 - node, 58
- openstack-baremetal-node-trait-add
 - command line option
 - node, 58
 - trait, 58
- openstack-baremetal-node-trait-list
 - command line option
 - column, 58
 - fit-width, 59
 - format, 58
 - max-width, 59
 - noindent, 59
 - print-empty, 59
 - quote, 58
 - sort-ascending, 59
 - sort-column, 59
 - sort-descending, 59
 - c, 58
 - f, 58
 - node, 59
- openstack-baremetal-node-trait-remove
 - command line option
 - all, 59
 - node, 59
 - trait, 59
- openstack-baremetal-node-undeploy
 - command line option
 - wait, 60
 - node, 60
- openstack-baremetal-node-unhold
 - command line option
 - node, 60
- openstack-baremetal-node-unrescue
 - command line option
 - wait, 60
 - node, 60
- openstack-baremetal-node-unset command
 - line option
 - automated-clean, 63
 - bios-interface, 62
 - boot-interface, 62
 - chassis-uuid, 62
 - conductor-group, 62
 - console-interface, 62
 - deploy-interface, 62
 - description, 63
 - driver-info, 62
 - extra, 61
 - firmware-interface, 62
 - inspect-interface, 62
 - instance-info, 62

```
--instance-name, 61
--instance-uuid, 61
--lessee, 63
--management-interface, 62
--name, 61
--network-data, 62
--network-interface, 62
--owner, 63
--parent-node, 63
--power-interface, 62
--property, 61
--protected, 63
--protected-reason, 63
--raid-interface, 62
--rescue-interface, 62
--resource-class, 61
--retired, 63
--retired-reason, 63
--shard, 63
--storage-interface, 62
--target-raid-config, 61
--vendor-interface, 62
node, 63
openstack-baremetal-node-validate
    command line option
    --column, 64
    --fit-width, 64
    --format, 63
    --max-width, 64
    --noindent, 64
    --print-empty, 64
    --quote, 64
    --sort-ascending, 64
    --sort-column, 64
    --sort-descending, 64
    -c, 64
    -f, 63
    node, 64
openstack-baremetal-node-vif-attach
    command line option
    --port-uuid, 64
    --portgroup-uuid, 64
    --vif-info, 64
    node, 65
    vif-id, 65
openstack-baremetal-node-vif-detach
    command line option
    node, 65
    vif-id, 65
openstack-baremetal-node-vif-list
    command line option
    --column, 65
    --fit-width, 66
    --format, 65
    --max-width, 65
    --noindent, 65
    --print-empty, 66
    --quote, 65
    --sort-ascending, 66
    --sort-column, 66
    --sort-descending, 66
    -c, 65
    -f, 65
    node, 66
openstack-baremetal-port-create
    command line option
    --category, 68
    --column, 67
    --description, 67
    --extra, 67
    --fit-width, 67
    --format, 66
    --is-smartnic, 67
    --local-link-connection, 67
    --max-width, 67
    --name, 67
    --node, 67
    --noindent, 67
    --physical-network, 67
    --port-group, 67
    --prefix, 67
    --print-empty, 67
    --pxe-enabled, 67
    --uuid, 67
    --vendor, 67
    -c, 67
    -f, 66
    address, 68
openstack-baremetal-port-delete
    command line option
    port, 68
openstack-baremetal-port-group-create
    command line option
    --address, 69
    --category, 69
    --column, 68
    --extra, 69
    --fit-width, 69
    --format, 68
    --max-width, 69
    --mode, 69
    --name, 69
```

```
--node, 69
--noindent, 68
--physical-network, 69
--prefix, 68
--print-empty, 69
--property, 69
--support-standalone-ports, 69
--unsupport-standalone-ports, 69
--uuid, 69
-c, 68
-f, 68
openstack-baremetal-port-group-delete
    command line option
port, 69
openstack-baremetal-port-group-list
    command line option
--address, 71
--column, 70
--fields, 71
--fit-width, 70
--format, 70
--limit, 71
--long, 71
--marker, 71
--max-width, 70
--node, 71
--noindent, 70
--print-empty, 70
--quote, 70
--shards, 71
--sort, 71
--sort-ascending, 70
--sort-column, 70
--sort-descending, 71
-c, 70
-f, 70
openstack-baremetal-port-group-set
    command line option
--address, 72
--category, 72
--extra, 72
--mode, 72
--name, 72
--node, 71
--physical-network, 72
--property, 72
--support-standalone-ports, 72
--unsupport-standalone-ports, 72
port, 72
openstack-baremetal-port-group-show
    command line option
--address, 73
--column, 73
--fields, 73
--fit-width, 73
--format, 72
--max-width, 73
--noindent, 73
--prefix, 73
--print-empty, 73
-c, 73
-f, 72
id, 73
openstack-baremetal-port-group-unset
    command line option
--address, 73
--category, 74
--extra, 73
--name, 73
--physical-network, 74
--property, 74
port, 74
openstack-baremetal-port-list command
    line option
--address, 75
--column, 74
--fields, 75
--fit-width, 74
--format, 74
--limit, 75
--long, 75
--marker, 75
--max-width, 74
--node, 75
--noindent, 74
--port-group, 75
--print-empty, 75
--quote, 74
--sort, 75
--sort-ascending, 75
--sort-column, 75
--sort-descending, 75
-c, 74
-f, 74
openstack-baremetal-port-set command
    line option
--address, 76
--category, 76
--description, 76
--extra, 76
--is-smartnic, 76
--local-link-connection, 76
```

```
--name, 76
--node, 76
--physical-network, 76
--port-group, 76
--pxe-disabled, 76
--pxe-enabled, 76
--vendor, 76
port, 76
openstack-baremetal-port-show command
    line option
    --address, 77
    --column, 77
    --fields, 77
    --fit-width, 77
    --format, 77
    --max-width, 77
    --noindent, 77
    --prefix, 77
    --print-empty, 77
    -c, 77
    -f, 77
    id, 77
openstack-baremetal-port-unset command
    line option
    --category, 78
    --description, 78
    --extra, 78
    --is-smartnic, 78
    --local-link-connection, 78
    --name, 78
    --physical-network, 78
    --port-group, 78
    --vendor, 78
    port, 78
openstack-baremetal-volume-connector-create
    command line option
    --column, 79
    --connector-id, 79
    --extra, 79
    --fit-width, 79
    --format, 79
    --max-width, 79
    --node, 79
    --noindent, 79
    --prefix, 79
    --print-empty, 79
    --type, 79
    --uuid, 79
    -c, 79
    -f, 79
openstack-baremetal-volume-connector-delete
    command line option
    volume, 80
openstack-baremetal-volume-connector-list
    command line option
    --column, 80
    --fields, 81
    --fit-width, 80
    --format, 80
    --limit, 81
    --long, 81
    --marker, 81
    --max-width, 80
    --node, 81
    --noindent, 80
    --print-empty, 80
    --quote, 80
    --sort, 81
    --sort-ascending, 81
    --sort-column, 81
    --sort-descending, 81
    -c, 80
    -f, 80
openstack-baremetal-volume-connector-set
    command line option
    --connector-id, 81
    --extra, 82
    --node, 81
    --type, 81
    volume, 82
openstack-baremetal-volume-connector-show
    command line option
    --column, 82
    --fields, 82
    --fit-width, 82
    --format, 82
    --max-width, 82
    --noindent, 82
    --prefix, 82
    --print-empty, 82
    -c, 82
    -f, 82
    id, 82
openstack-baremetal-volume-connector-unset
    command line option
    --extra, 83
    volume, 83
openstack-baremetal-volume-target-create
    command line option
    --boot-index, 84
    --column, 83
    --extra, 84
```

```

--fit-width, 84
--format, 83
--max-width, 83
--node, 84
--noindent, 83
--prefix, 83
--print-empty, 84
--property, 84
--type, 84
--uuid, 84
--volume-id, 84
-c, 83
-f, 83
openstack-baremetal-volume-target-delete
    command line option
    volume, 84
openstack-baremetal-volume-target-list
    command line option
--column, 85
--fields, 86
--fit-width, 85
--format, 85
--limit, 85
--long, 86
--marker, 86
--max-width, 85
--node, 85
--noindent, 85
--print-empty, 85
--quote, 85
--sort, 86
--sort-ascending, 85
--sort-column, 85
--sort-descending, 85
-c, 85
-f, 85
openstack-baremetal-volume-target-set
    command line option
--boot-index, 86
--extra, 86
--node, 86
--property, 86
--type, 86
--volume-id, 86
    volume, 86
openstack-baremetal-volume-target-show
    command line option
--column, 87
--fields, 87
--fit-width, 87
--format, 87
--max-width, 87
--noindent, 87
--prefix, 87
--print-empty, 87
-c, 87
-f, 87
id, 87
openstack-baremetal-volume-target-unset
    command line option
--extra, 88
--property, 88
    volume, 88
os_ironic_api_version (ironic-
    client.common.http.VersionNegotiationMixin
    attribute), 114

```

P

```

PassthruCallBaremetalDriver (class in iron-
    icclient.osc.v1.baremetal_driver), 130
PassthruCallBaremetalNode (class in ironic-
    client.osc.v1.baremetal_node), 145
PassthruListBaremetalDriver (class in iron-
    icclient.osc.v1.baremetal_driver), 131
PassthruListBaremetalNode (class in ironic-
    client.osc.v1.baremetal_node), 145
PaymentRequired, 104
poll() (in module ironicclient.common.utils), 118
port
    openstack-baremetal-port-delete
        command line option, 68
    openstack-baremetal-port-group-delete
        command line option, 69
    openstack-baremetal-port-group-set
        command line option, 72
    openstack-baremetal-port-group-unset
        command line option, 74
    openstack-baremetal-port-set
        command line option, 76
    openstack-baremetal-port-unset
        command line option, 78
Port (class in ironicclient.v1.port), 197
Portgroup (class in ironicclient.v1.portgroup),
    198
PortgroupManager (class in ironic-
    client.v1.portgroup), 198
PortManager (class in ironicclient.v1.port), 197
POWER_STATE (ironic-
    client.osc.v1.baremetal_node.PowerBaremetalNode
    attribute), 146
POWER_STATE (ironic-
    client.osc.v1.baremetal_node.PowerOffBaremetalNode

```

attribute), 146

POWER_STATE (ironic-client.osc.v1.baremetal_node.PowerOnBaremetalNode attribute), 147

PowerBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 146

PowerOffBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 146

PowerOnBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 147

PreconditionFailed, 105

properties() (ironic-client.v1.driver.DriverManager method), 181

ProvideBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 147

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.AbortBaremetalNode attribute), 135

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.AdoptBaremetalNode attribute), 136

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.CleanBaremetalNode attribute), 138

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.DeployBaremetalNode attribute), 140

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.InspectBaremetalNode attribute), 141

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.ManageBaremetalNode attribute), 144

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.ProvideBaremetalNode attribute), 147

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.ProvisionStateBaremetalNode attribute), 147

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.RebuildBaremetalNode attribute), 148

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.RescueBaremetalNode attribute), 149

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.ServiceBaremetalNode attribute), 150

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.UndeployBaremetalNode attribute), 151

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.UnholdBaremetalNode attribute), 151

PROVISION_STATE (ironic-client.osc.v1.baremetal_node.UnrescueBaremetalNode attribute), 152

PROVISION_STATES (ironic-client.osc.v1.baremetal_node.ListBaremetalNode attribute), 141

ProvisionStateBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 147

ProvisionStateWithWait (class in ironic-client.osc.v1.baremetal_node), 147

ProxyAuthenticationRequired, 106

R

raidLogicalDiskProperties() (ironic-client.v1.driver.DriverManager method), 181

raidRequest() (ironic-client.common.http.SessionClient method), 114

RebuildBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 148

RebuildBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 148

remove_all_traits() (ironic-client.v1.node.NodeManager method), 190

remove_all_traits() (ironic-client.v1.runbook.RunbookManager method), 204

remove_trait() (ironic-client.v1.node.NodeManager method), 190

remove_trait() (ironic-client.v1.runbook.RunbookManager method), 204

RemoveTraitBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 148

RemoveTraitBaremetalRunbook (class in ironic-client.osc.v1.baremetal_runbook), 160

ReplaceLatestVersion (class in ironic-client.osc.plugin), 169

RequestedRangeNotSatisfiable, 108

RequestEntityTooLarge, 107

RequestTimeout, 107

RequestUriTooLong, 107

RescueBaremetalNode (class in ironic-client.osc.v1.baremetal_node), 149

Resource	(class in ironic-client.common.apiclient.base), 93	retrieve_data()	(in module ironic-client.common.filecache), 114
Resource	(class in ironicclient.common.base), 113	Runbook	(class in ironicclient.v1.runbook), 203
Resource	(class in ironic-client.v1.resource_fields), 201	RunbookManager	(class in ironic-client.v1.runbook), 203
resource_class	(ironic-client.common.base.Manager property), 113	S	
resource_class	(ironic-client.v1.allocation.AllocationManager attribute), 171	save_data()	(in module ironic-client.common.filecache), 114
resource_class	(ironic-client.v1.chassis.ChassisManager attribute), 174	SecurebootOffBaremetalNode	(class in ironic-client.osc.v1.baremetal_node), 149
resource_class	(ironic-client.v1.conductor.ConductorManager attribute), 176	SecurebootOnBaremetalNode	(class in ironic-client.osc.v1.baremetal_node), 150
resource_class	(ironic-client.v1.deploy_template.DeployTemplateManager attribute), 180	ServiceBaremetalNode	(class in ironic-client.osc.v1.baremetal_node), 150
resource_class	(ironic-client.v1.driver.DriverManager attribute), 181	ServiceUnavailable	110
resource_class	(ironic-client.v1.events.EventManager attribute), 182	SessionClient	(class in ironic-client.common.http), 114
resource_class	(ironic-client.v1.inspection_rule.InspectionRuleManager attribute), 183	set_boot_device()	(ironic-client.v1.node.NodeManager method), 191
resource_class	(ironic-client.v1.node.NodeManager attribute), 191	set_boot_mode()	(ironic-client.v1.node.NodeManager method), 191
resource_class	(ironic-client.v1.port.PortManager attribute), 198	set_console_mode()	(ironic-client.v1.node.NodeManager method), 191
resource_class	(ironic-client.v1.portgroup.PortgroupManager attribute), 201	set_loaded()	(ironic-client.common.apiclient.base.Resource method), 93
resource_class	(ironic-client.v1.runbook.RunbookManager attribute), 205	set_maintenance()	(ironic-client.v1.node.NodeManager method), 191
resource_class	(ironic-client.v1.shard.ShardManager attribute), 205	set_power_state()	(ironic-client.v1.node.NodeManager method), 192
resource_class	(ironic-client.v1.volume_connector.VolumeConnectorManager attribute), 206	set_provision_state()	(ironic-client.v1.node.NodeManager method), 192
resource_class	(ironic-client.v1.volume_target.VolumeTargetManager attribute), 208	set_secure_boot()	(ironic-client.v1.node.NodeManager method), 193
		set_target_raid_config()	(ironic-client.v1.node.NodeManager method), 194
		set_traits()	(ironic-client.v1.node.NodeManager method), 194
		SetBaremetalAllocation	(class in ironic-client.osc.v1.baremetal_allocation), 120

`SetBaremetalChassis` (class in `ironic-client.osc.v1.baremetal_chassis`), 123

`SetBaremetalDeployTemplate` (class in `ironic-client.osc.v1.baremetal_deploy_template`), 127

`SetBaremetalInspectionRule` (class in `ironic-client.osc.v1.baremetal_inspection_rule`), 133

`SetBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 150

`SetBaremetalPort` (class in `ironic-client.osc.v1.baremetal_port`), 155

`SetBaremetalPortGroup` (class in `ironic-client.osc.v1.baremetal_portgroup`), 157

`SetBaremetalRunbook` (class in `ironic-client.osc.v1.baremetal_runbook`), 161

`SetBaremetalVolumeConnector` (class in `ironic-client.osc.v1.baremetal_volume_connector`), 164

`SetBaremetalVolumeTarget` (class in `ironic-client.osc.v1.baremetal_volume_target`), 167

setting

- `openstack-baremetal-node-bios-settings` command line option, 32

`Shard` (class in `ironicclient.v1.shard`), 205

`ShardManager` (class in `ironicclient.v1.shard`), 205

`ShowBaremetalAllocation` (class in `ironic-client.osc.v1.baremetal_allocation`), 121

`ShowBaremetalChassis` (class in `ironic-client.osc.v1.baremetal_chassis`), 124

`ShowBaremetalConductor` (class in `ironic-client.osc.v1.baremetal_conductor`), 125

`ShowBaremetalDeployTemplate` (class in `ironic-client.osc.v1.baremetal_deploy_template`), 128

`ShowBaremetalDriver` (class in `ironic-client.osc.v1.baremetal_driver`), 131

`ShowBaremetalInspectionRule` (class in `ironic-client.osc.v1.baremetal_inspection_rule`), 134

`ShowBaremetalNode` (class in `ironic-client.osc.v1.baremetal_node`), 151

`ShowBaremetalPort` (class in `ironic-client.osc.v1.baremetal_port`), 155

`ShowBaremetalPortGroup` (class in `ironic-client.osc.v1.baremetal_portgroup`), 157

`ShowBaremetalRunbook` (class in `ironic-client.osc.v1.baremetal_runbook`), 161

`ShowBaremetalVolumeConnector` (class in `ironic-client.osc.v1.baremetal_volume_connector`), 164

`ShowBaremetalVolumeTarget` (class in `ironic-client.osc.v1.baremetal_volume_target`), 168

`ShowOne` (class in `ironicclient.osc.command`), 169

`sort_dir` (`ironic-client.common.utils.ListCommandArgs` attribute), 115

`sort_fields` (`ironic-client.v1.resource_fields.Resource` property), 203

`sort_key` (`ironic-client.common.utils.ListCommandArgs` attribute), 115

`sort_labels` (`ironic-client.v1.resource_fields.Resource` property), 203

`show_and_deserialize()` (in module `ironic-client.common.utils`), 119

`states()` (`ironicclient.v1.node.NodeManager` method), 194

`StateTransitionFailed`, 209

`StateTransitionTimeout`, 209

T

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.CreateBaremetalAllocation` method), 119

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.DeleteBaremetalAllocation` method), 120

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.ListBaremetalAllocations` method), 120

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.SetBaremetalAllocation` method), 121

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.ShowBaremetalAllocation` method), 121

`take_action()` (`ironic-client.osc.v1.baremetal_allocation.UnsetBaremetalAllocation` method), 121

method), 122

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.CreateBaremetalChassis
method), 122 method), 130

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.DeleteBaremetalChassis
method), 123 method), 131

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.ListBaremetalChassis
method), 123 method), 131

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.SetBaremetalChassis
method), 124 method), 132

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.ShowBaremetalChassis
method), 124 method), 132

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_chassis.UnsetBaremetalChassis
method), 124 method), 133

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_conductor.ListBaremetalConductors
method), 125 method), 133

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_conductor.ShowBaremetalConductor
method), 125 method), 134

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_create.CreateBaremetal
method), 126 client.osc.v1.baremetal_inspection_rule.ShowBaremetalInspectionRule
method), 134

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.CreateBaremetalDeployTemplate
method), 126 client.osc.v1.baremetal_inspection_rule.UnsetBaremetalInspectionRule
method), 135

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.DeleteBaremetalDeployTemplate
method), 127 client.osc.v1.baremetal_node.AddTraitBaremetalNode
method), 136

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.ListBaremetalDeployTemplates
method), 127 client.osc.v1.baremetal_node.BIOSSettingShowBaremetalNode
method), 136

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.SetBaremetalDeployTemplate
method), 128 client.osc.v1.baremetal_node.BootdeviceSetBaremetalNode
method), 137

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.ShowBaremetalDeployTemplate
method), 128 client.osc.v1.baremetal_node.BootdeviceShowBaremetalNode
method), 137

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_deploy_template.UnsetBaremetalDeployTemplate
method), 129 client.osc.v1.baremetal_node.BootmodeSetBaremetalNode
method), 138

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_driver.ListBaremetalDriver
method), 129 client.osc.v1.baremetal_node.ConsoleDisableBaremetalNode
method), 138

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_driver.ListBaremetalDriverProperties
method), 129 client.osc.v1.baremetal_node.ConsoleEnableBaremetalNode
method), 138

<code>method), 139</code>	<code>method), 146</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.ConsoleShowBaremetalNode</code> <code>method), 139</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.ProvisionStateBaremetalNode</code> <code>method), 147</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.CreateBaremetalNode</code> <code>method), 140</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.ProvisionStateWithWait</code> <code>method), 148</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.DeleteBaremetalNode</code> <code>method), 140</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.RebootBaremetalNode</code> <code>method), 148</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.InjectNmiBaremetalNode</code> <code>method), 141</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.RemoveTraitBaremetalNode</code> <code>method), 149</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.ListBaremetalNode</code> <code>method), 142</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.SecurebootOffBaremetalNode</code> <code>method), 149</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.ListBIOSSettingBaremetalNode</code> <code>method), 141</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.SecurebootOnBaremetalNode</code> <code>method), 150</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.ListFirmwareComponentsBaremetalNode</code> <code>method), 142</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.SetBaremetalNode</code> <code>method), 151</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.ListTraitsBaremetalNode</code> <code>method), 142</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.ShowBaremetalNode</code> <code>method), 151</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.MaintenanceSetBaremetalNode</code> <code>method), 143</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.UnsetBaremetalNode</code> <code>method), 152</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.MaintenanceUnsetBaremetalNode</code> <code>method), 143</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.ValidateBaremetalNode</code> <code>method), 152</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.NodeChildrenList</code> <code>method), 144</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.VifAttachBaremetalNode</code> <code>method), 153</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.NodeHistoryEventGet</code> <code>method), 144</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.VifDetachBaremetalNode</code> <code>method), 153</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.NodeHistoryList</code> <code>method), 145</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_node.VifListBaremetalNode</code> <code>method), 153</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.NodeInventorySave</code> <code>method), 145</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_port.CreateBaremetalPort</code> <code>method), 154</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.PassthruCallBaremetalNode</code> <code>method), 145</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_port.DeleteBaremetalPort</code> <code>method), 154</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.PassthruListBaremetalNode</code> <code>method), 146</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_port.ListBaremetalPort</code> <code>method), 154</code>
<code>take_action()</code> <code>client.osc.v1.baremetal_node.PowerBaremetalNode</code>	<code>(ironic- take_action()</code> <code>client.osc.v1.baremetal_port.SetBaremetalPort</code>

method), 155

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_port.ShowBaremetalPort client.osc.v1.baremetal_shard.ListBaremetalShard
method), 155 method), 162

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_port.UnsetBaremetalPort client.osc.v1.baremetal_volume_connector.CreateBaremetalVolume
method), 155 method), 163

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.CreateBaremetalPortGroup client.osc.v1.baremetal_volume_connector.DeleteBaremetalVolume
method), 156 method), 163

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.DeleteBaremetalPortGroup client.osc.v1.baremetal_volume_connector.ListBaremetalVolumes
method), 156 method), 164

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.ListBaremetalPortGroups client.osc.v1.baremetal_volume_connector.SetBaremetalVolume
method), 157 method), 164

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.SetBaremetalPortGroup client.osc.v1.baremetal_volume_connector.ShowBaremetalVolume
method), 157 method), 165

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.ShowBaremetalPortGroup client.osc.v1.baremetal_volume_connector.UnsetBaremetalVolume
method), 158 method), 165

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_portgroup.UnsetBaremetalPortGroup client.osc.v1.baremetal_volume_target.CreateBaremetalVolume
method), 158 method), 166

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_runbook.AddTraitBaremetalRunbook client.osc.v1.baremetal_volume_target.DeleteBaremetalVolume
method), 158 method), 166

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_runbook.CreateBaremetalRunbook client.osc.v1.baremetal_volume_target.ListBaremetalVolumes
method), 159 method), 167

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_runbook.DeleteBaremetalRunbook client.osc.v1.baremetal_volume_target.SetBaremetalVolume
method), 159 method), 167

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_runbook.ListBaremetalRunbooks client.osc.v1.baremetal_volume_target.ShowBaremetalVolume
method), 160 method), 168

take_action() (ironic- take_action() (ironic-
client.osc.v1.baremetal_runbook.ListTraitsBaremetalRunbook client.osc.v1.baremetal_volume_target.UnsetBaremetalVolume
method), 160 method), 168

take_action() (ironic- tmpdir() (in module ironicclient.common.utils),
client.osc.v1.baremetal_runbook.RemoveTraitBaremetalRunbook
method), 161 template

take_action() (ironic- openstack-baremetal-deploy-template-delete
client.osc.v1.baremetal_runbook.SetBaremetalRunbook command line option, 20
method), 161 openstack-baremetal-deploy-template-set

take_action() (ironic- command line option, 22
client.osc.v1.baremetal_runbook.ShowBaremetalRunbook openstack-baremetal-deploy-template-show
method), 161 command line option, 23

take_action() (ironic- openstack-baremetal-deploy-template-unset
client.osc.v1.baremetal_runbook.UnsetBaremetalRunbook command line option, 23

- `to_dict()` (*ironic-client.common.apiclient.base.Resource method*), 93
- `to_dict()` (*ironicclient.common.base.Resource method*), 113
- `trait`
- `openstack-baremetal-node-add-trait` command line option, 29
 - `openstack-baremetal-node-remove-trait` command line option, 51
 - `openstack-baremetal-node-trait-add` command line option, 58
 - `openstack-baremetal-node-trait-remove` command line option, 59
- ## U
- `Unauthorized`, 110
- `UndeployBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 151
- `UnholdBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 151
- `UnprocessableEntity`, 111
- `UnrescueBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 151
- `UnsetBaremetalAllocation` (class in *ironic-client.osc.v1.baremetal_allocation*), 121
- `UnsetBaremetalChassis` (class in *ironic-client.osc.v1.baremetal_chassis*), 124
- `UnsetBaremetalDeployTemplate` (class in *ironic-client.osc.v1.baremetal_deploy_template*), 129
- `UnsetBaremetalInspectionRule` (class in *ironic-client.osc.v1.baremetal_inspection_rule*), 135
- `UnsetBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 152
- `UnsetBaremetalPort` (class in *ironic-client.osc.v1.baremetal_port*), 155
- `UnsetBaremetalPortGroup` (class in *ironic-client.osc.v1.baremetal_portgroup*), 158
- `UnsetBaremetalRunbook` (class in *ironic-client.osc.v1.baremetal_runbook*), 161
- `UnsetBaremetalVolumeConnector` (class in *ironic-client.osc.v1.baremetal_volume_connector*), 165
- `UnsetBaremetalVolumeTarget` (class in *ironic-client.osc.v1.baremetal_volume_target*), 168
- `UnsupportedMediaType`, 111
- `UnsupportedVersion`, 112
- `update()` (*ironic-client.v1.allocation.AllocationManager method*), 171
- `update()` (*ironic-client.v1.chassis.ChassisManager method*), 174
- `update()` (*ironic-client.v1.deploy_template.DeployTemplateManager method*), 180
- `update()` (*ironicclient.v1.driver.DriverManager method*), 181
- `update()` (*ironic-client.v1.inspection_rule.InspectionRuleManager method*), 183
- `update()` (*ironicclient.v1.node.NodeManager method*), 194
- `update()` (*ironicclient.v1.port.PortManager method*), 198
- `update()` (*ironic-client.v1.portgroup.PortgroupManager method*), 201
- `update()` (*ironic-client.v1.runbook.RunbookManager method*), 205
- `update()` (*ironic-client.v1.volume_connector.VolumeConnectorManager method*), 207
- `update()` (*ironic-client.v1.volume_target.VolumeTargetManager method*), 208
- ## V
- `validate()` (*ironicclient.v1.node.NodeManager method*), 194
- `ValidateBaremetalNode` (class in *ironic-client.osc.v1.baremetal_node*), 152
- `ValidationError`, 112
- `vendor_passthru()` (*ironic-client.v1.driver.DriverManager method*), 181
- `vendor_passthru()` (*ironic-client.v1.node.NodeManager method*), 195
- `VersionNegotiationMixin` (class in *ironic-client.common.http*), 114
- `vif_attach()` (*ironic-client.v1.node.NodeManager method*), 195
- `vif_detach()` (*ironic-*

- client.v1.node.NodeManager* method),
195
- vif_list()* (*ironicclient.v1.node.NodeManager*
method), 195
- vif-id*
 - openstack-baremetal-node-vif-attach*
command line option, 65
 - openstack-baremetal-node-vif-detach*
command line option, 65
- VifAttachBaremetalNode* (class in *ironic-*
client.osc.v1.baremetal_node), 152
- VifDetachBaremetalNode* (class in *ironic-*
client.osc.v1.baremetal_node), 153
- VifListBaremetalNode* (class in *ironic-*
client.osc.v1.baremetal_node), 153
- volume*
 - openstack-baremetal-volume-connector-delete*
command line option, 80
 - openstack-baremetal-volume-connector-set*
command line option, 82
 - openstack-baremetal-volume-connector-unset*
command line option, 83
 - openstack-baremetal-volume-target-delete*
command line option, 84
 - openstack-baremetal-volume-target-set*
command line option, 86
 - openstack-baremetal-volume-target-unset*
command line option, 88
- VolumeConnector* (class in *ironic-*
client.v1.volume_connector), 205
- VolumeConnectorManager* (class in *ironic-*
client.v1.volume_connector), 206
- VolumeTarget* (class in *ironic-*
client.v1.volume_target), 207
- VolumeTargetManager* (class in *ironic-*
client.v1.volume_target), 207

W

- wait()* (*ironicclient.v1.allocation.AllocationManager*
method), 172
- wait_for_provision_state()* (*ironic-*
client.v1.node.NodeManager method),
196
- with_retries()* (in module *ironic-*
client.common.http), 115