caso Documentation

Spanish National Research Council (CSIC)

Mar 09, 2020

Contents

`cASO` is an accounting reporter (currently supports Cloud Accounting Usage Records) for OpenStack deployments.
`cASO` gets usage information from nova or ceilometer public APIs (no access to DB is required) and can generate
valid output for Apel SSM or logstash.

Contents:

Installation
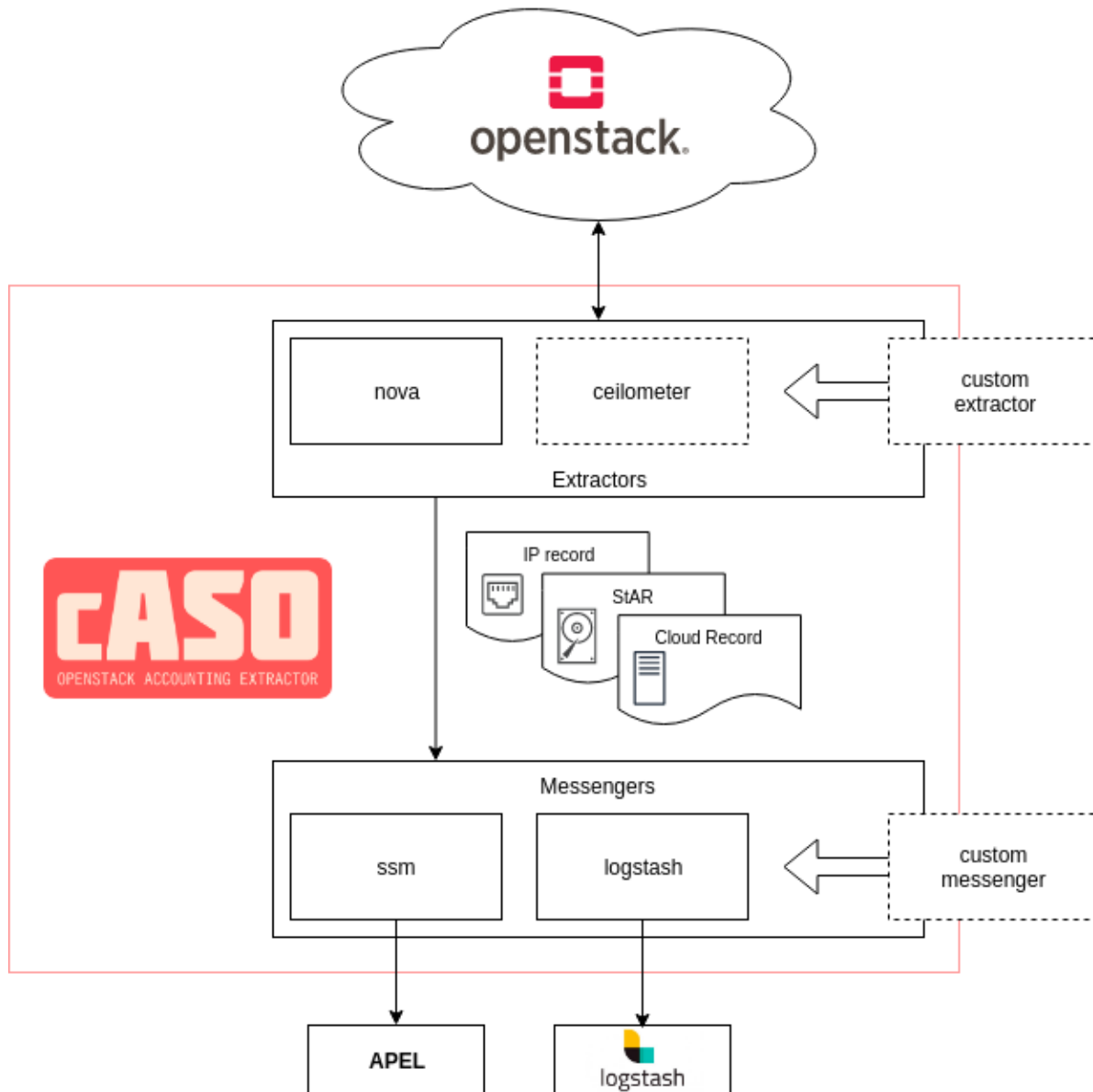
1.1 Pre-requisites

If you are planning to use `cASO` for generating accounting records for EGI, you will need a valid APEL/SSM configuration. Follow the documentation available at the EGI.eu Federated Cloud documentation in order to set it up.

1.2 Installation

The best way to install cASO and have the most up to date version is using the repositories and packages provided in the EGI AppDB:

> https://appdb.egi.eu/store/software/caso

1.2.1 Manual installation

Even the reccomended method is to use a package from the EGI AppDB, it is possible to install it from the Python Pacakge Index as follows:

```
$ pip install caso
```

Or you can install it on a virtualenv:

```
$ virtualenv --python python3 caso
$ source caso/bin/activate
(caso) $ pip install caso
```

Configuration

2.1 OpenStack Configuration

Apart from configuring cASO, several actions need to be performed in your OpenStack installation in order to be able to extract accounting records.

2.1.1 User credentials (required)

In the next section you will configure an OpenStack Keystone credentials in order to extract the records. The cASO user has to be a member of each of the projects (another option is to convert that user in an administrator, but the former option is a safer approach) for which it is extracting the accounting. Otherwise, `cASO` will not be able to get the usages and will fail.

In order to do so, we are going to setup a new role `accounting` a new user `accounting`, adding it to each of the projects with that role:

```
openstack role create accounting
openstack user create --password <password> accounting
# For each of the projects, add the user with the accounting role
openstack role add --user accounting --project <project> accounting
```

Moreover, this user needs access to Keystone so as to extract the users information. In this case, we can can grant the user just the rights for listing the users adding the appropriate rules in your `/etc/keystone/policy.json` as follows. Replace the line:

```
"identity:list_users": "rule:admin_required",
```

with:

```
"identity:list_users": "rule:admin_required or role:accounting",
```

Recent Keystone versions leverage a `/etc/keystone/policy-yaml` file, if this is your case, substitute the line:

```
"identity:list_users": "rule:admin_required"
```

with:

```
"identity:list_users": "rule:admin_required or role:accounting"
```

2.1.2 Publishing benchmark information for OpenStack flavors (optional)

Starting with the V0.4 of the accounting record it is possible to publish benchmark information. In order to do so, you need to add this information to the flavor properties and configure caso to retrieve this information. There are two different values that need to be added to the flavor:

- The benchmark name, indicated with the `accounting:benchmark_name` flavor property.

- The benchmark value, indicated with the `accounting:benchmark_value` flavor property.

For example, if you are using HEPSPEC06 and the benchmark value is `99` for the flavor `m1.foo`, the benchmark information is configured as follows:

```
openstack flavor set --property accounting:benchmark_name="HEPSPEC06" --property
→accounting:benchmark_value=99 m1.foo
```

Using different keys to specify bechmark information

If you do not want to use cASO's default flavor properties `accounting:benchmark_name` and `accounting:benchmark_value` (for example because you are using different benchmark types and values) you can specify which properties `cASO` should look for by using the `benchmark_name_key` `benchkark_value_key` in the configuration file.

Important: Please note that there is an OpenStack scheduler filter that removes hosts based on flavor properties. In order to not interfere with the behaviour of this filter you must prefix the property with a `scope:` so that cASO's properties are not taken into account for this filtering. When adding these properties in cASO's configuration file, please include the complete name (i.e. `scope:property`).

2.2 cASO configuration

`cASO` uses a config file (default at `/etc/caso/caso.conf`) with several sections. A sample file is available at `etc/caso/caso.conf.sample`.

2.2.1 `[DEFAULT]` section

The `[DEFAULT]` section configures the basic behavior of `cASO`. The sample config file (`/etc/caso/caso.conf.sample`) includes a description of every option. You should check at least the following options:

- `extractor` (default value: `nova`), specifies which extractor to use for getting the data. The following APIs are supported: `ceilomenter` and `nova`. Both should generate equivalent information.

- `site_name` (default value: `<None>`). Name of the site as defined in GOCDB.

- `service_name` (default value: `$site_name`). Name of the service within a site. This is used if you have several endpoints within your site.

- `projects` (list value, default empty). List of the projects to extract records from.

- `messengers` (list, default: `noop`). List of the messengers to publish data to. Records will be pushed to all these messengers, in order. Valid messengers shipped with cASO are:

    - `ssm` for publishing APEL V0.4 records.

    - `logstash` for publishing to Logstash.

    - `noop` do nothing at all.

  Note that there might be other messengers available in the system if they are registered into the `caso.messenger` entry point namespace.

- `mapping_file` (default: `/etc/caso/voms.json`). File containing the mapping from VOs to local projects as configured in Keystone-VOMS, in the form:

```
{
    "VO": {
        "projects": ["foo", "bar"],
    }
}
```

- `benchmark_name_key` and `benchmark_value_key`. These two configuration options are used by cASO to retrieve the benchmark information form the OpenStack flavors.

2.2.2 `[keystone_auth]` section

This section is used to specify the authentication credentials to be used to connect to the OpenStack APIs. cASO leverages the OpenStack keystoneauth library for authentication, so that it is possible to use any authentication plugin that is available there (so starting on version 1.0 of cASO it is possible to use the Keystone V3 API).

Important: You need to specify the `auth_type` that you want to use (normally `v3password` is a good choice.

For an exhaustive list of available plugins please refer to the keystoneauth documentation.

2.2.3 `[ssm]` section

Options defined here configure the SSM messenger. There is only one option at the moment:

- `output_path` (default: `/var/spool/apel/outgoing/openstack`), directory to put the generated SSM records. APEL/SSM should be configured to take records from that directory.

2.2.4 `[logstash]` section

Options defined here configure the logstash messenger. Available options:

- `host` (default: `localhost`), host of Logstash server.

- `port` (default: `5000`), Logstash server port.

2.2.5 Other cASO configuration options

For an exhaustive list of the defined options, please check the following page:

---

cASO configuration file

oslo.log: DEFAULT

debug

>>> **Type** boolean
>>>
>>> **Default** `false`
>>>
>>> **Mutable** This option can be changed without restarting.

If set to true, the logging level will be set to DEBUG instead of the default INFO level.

log_config_append

>>> **Type** string
>>>
>>> **Default** `<None>`
>>>
>>> **Mutable** This option can be changed without restarting.

The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, log-date-format).

Table 1: Deprecated Variations

Group	Name
DEFAULT	log-config
DEFAULT	log_config

log_date_format

>>> **Type** string
>>>
>>> **Default** `%Y-%m-%d %H:%M:%S`

Defines the format string for %(asctime)s in log records. Default: the value above . This option is ignored if log_config_append is set.

log_file

>>> **Type** string
>>>
>>> **Default** `<None>`

(Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.

Table 2: Deprecated Variations

Group	Name
DEFAULT	logfile

log_dir

>>> **Type** string
>>>
>>> **Default** `<None>`

(Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.

Table 3: Deprecated Variations

Group	Name
DEFAULT	logdir

watch_log_file

> **Type** boolean
>
> **Default** `false`

Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log_file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

use_syslog

> **Type** boolean
>
> **Default** `false`

Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.

use_journal

> **Type** boolean
>
> **Default** `false`

Enable journald for logging. If running in a systemd environment you may wish to enable journal support. Doing so will use the journal native protocol which includes structured metadata in addition to log messages.This option is ignored if log_config_append is set.

syslog_log_facility

> **Type** string
>
> **Default** `LOG_USER`

Syslog facility to receive log lines. This option is ignored if log_config_append is set.

use_json

> **Type** boolean
>
> **Default** `false`

Use JSON formatting for logging. This option is ignored if log_config_append is set.

use_stderr

> **Type** boolean
>
> **Default** `false`

Log output to standard error. This option is ignored if log_config_append is set.

use_eventlog

> **Type** boolean
>
> **Default** `false`

Log output to Windows Event Log.

log_rotate_interval

> **Type** integer
>
> **Default** `1`

The amount of time before the log files are rotated. This option is ignored unless log_rotation_type is setto "interval".

log_rotate_interval_type

> **Type** string
>
> **Default** `days`
>
> **Valid Values** Seconds, Minutes, Hours, Days, Weekday, Midnight

Rotation interval type. The time of the last file change (or the time when the service was started) is used when scheduling the next rotation.

max_logfile_count

> **Type** integer
>
> **Default** `30`

Maximum number of rotated log files.

max_logfile_size_mb

> **Type** integer
>
> **Default** `200`

Log file maximum size in MB. This option is ignored if "log_rotation_type" is not set to "size".

log_rotation_type

> **Type** string
>
> **Default** `none`
>
> **Valid Values** interval, size, none

Log rotation type.

Possible values

interval Rotate logs at predefined time intervals.

size Rotate logs once they reach a predefined size.

none Do not rotate log files.

logging_context_format_string

> **Type** string
>
> **Default** `%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s`

Format string to use for log messages with context. Used by oslo_log.formatters.ContextFormatter

logging_default_format_string

> **Type** string

> **Default** `%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-] %(instance)s%(message)s`

Format string to use for log messages when context is undefined. Used by oslo_log.formatters.ContextFormatter

logging_debug_format_suffix

> **Type** string

> **Default** `%(funcName)s %(pathname)s:%(lineno)d`

Additional data to append to log message when logging level for the message is DEBUG. Used by oslo_log.formatters.ContextFormatter

logging_exception_prefix

> **Type** string

> **Default** `%(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s`

Prefix each line of exception output with this format. Used by oslo_log.formatters.ContextFormatter

logging_user_identity_format

> **Type** string

> **Default** `%(user)s %(tenant)s %(domain)s %(user_domain)s %(project_domain)s`

Defines the format string for %(user_identity)s that is used in logging_context_format_string. Used by oslo_log.formatters.ContextFormatter

default_log_levels

> **Type** list

> **Default** `amqp=WARN,amqplib=WARN,boto=WARN,qpid=WARN,sqlalchemy=WARN, suds=INFO,oslo.messaging=INFO,oslo_messaging=INFO, iso8601=WARN,requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN,websocket=WARN,requests. packages.urllib3.util.retry=WARN,urllib3.util.retry=WARN, keystonemiddleware=WARN,routes.middleware=WARN,stevedore=WARN, taskflow=WARN,keystoneauth=WARN,oslo.cache=INFO, oslo_policy=INFO,dogpile.core.dogpile=INFO`

List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.

publish_errors

> **Type** boolean

> **Default** `false`

Enables or disables publication of error events.

instance_format

> **Type** string

> **Default** `"[instance:  %(uuid)s] "`

The format for an instance that is passed with the log message.

instance_uuid_format

> **Type** string

Default `"[instance:  %(uuid)s] "`

The format for an instance UUID that is passed with the log message.

rate_limit_interval

> **Type** integer
>
> **Default** 0

Interval, number of seconds, of log rate limiting.

rate_limit_burst

> **Type** integer
>
> **Default** 0

Maximum number of logged messages per rate_limit_interval.

rate_limit_except_level

> **Type** string
>
> **Default** `CRITICAL`

Log level name used by rate limiting: CRITICAL, ERROR, INFO, WARNING, DEBUG or empty string. Logs with level greater or equal to rate_limit_except_level are not filtered. An empty string means that all levels are filtered.

fatal_deprecations

> **Type** boolean
>
> **Default** `false`

Enables or disables fatal status of deprecations.

oslo.config: DEFAULT

config_file

> **Type** list of filenames
>
> **Default** `~/.project/project.conf,~/project.conf,/etc/project/project.`
> `conf,/etc/project.conf`

Path to a config file to use. Multiple config files can be specified, with values in later files taking precedence. Defaults to the value above. This option must be set from the command-line.

config_dir

> **Type** list of directory names
>
> **Default** `~/.project/project.conf.d/,~/project.conf.d/,/etc/project/`
> `project.conf.d/,/etc/project.conf.d/`

Path to a config directory to pull *.conf* files from. This file set is sorted, so as to provide a predictable parse order if individual options are over-ridden. The set is parsed after the file(s) specified via previous –config-file, arguments hence over-ridden options in the directory take precedence. This option must be set from the command-line.

config_source

> **Type** list

> **Default** `u''`

Lists configuration groups that provide more details for accessing configuration settings from locations other than local files.

driver

> **Type** string
>
> **Default** `remote_file`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The name of the driver that can load this configuration source.

uri

> **Type** URI
>
> **Default** `https://example.com/my-configuration.ini`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Required option with the URI of the extra configuration file's location.

ca_path

> **Type** string
>
> **Default** `/etc/ca-certificates`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The path to a CA_BUNDLE file or directory with certificates of trusted CAs.

client_cert

> **Type** string
>
> **Default** `/etc/ca-certificates/service-client-keystore`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Client side certificate, as a single file path containing either the certificate only or the private key and the certificate.

client_key

> **Type** string
>
> **Default** `<None>`

Client side private key, in case client_cert is specified but does not includes the private key.

caso: DEFAULT

messengers

> **Type** list
>
> **Default** `noop`

List of messengers that will dispatch records. valid values are ssm,noop,logstash. You can specify more than one messenger.

spooldir

> **Type** string
>
> **Default** `/var/spool/caso`

Spool directory.

lock_path

> **Type** string
>
> **Default** `$spooldir`

Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable CASO_LOCK_PATH or $spooldir

dry_run

> **Type** boolean
>
> **Default** `false`

Extract records but do not push records to SSM. This will not update the last run date.

Table 4: Deprecated Variations

Group	Name
DEFAULT	dry_run

site_name

> **Type** string
>
> **Default** `<None>`

Site name as in GOCDB.

service_name

> **Type** string
>
> **Default** `$site_name`

Service name within the site

benchmark_name_key

> **Type** string
>
> **Default** `accounting:benchmark_type`

Metadata key used to retrieve the benchmark type from the flavor properties.

benchmark_value_key

> **Type** string
>
> **Default** `accounting:benchmark_value`

Metadata key used to retrieve the benchmark value from the flavor properties.

projects

> **Type** list

> **Default** `u''`

List of projects to extract accounting records from.

Table 5: Deprecated Variations

Group	Name
DEFAULT	tenants

`mapping_file`

> **Type** string
>
> **Default** `/etc/caso/voms.json`

File containing the VO <-> project mapping as used in Keystone-VOMS.

Table 6: Deprecated Variations

Group	Name
extractor	mapping_file

`extract_to`

> **Type** string
>
> **Default** `<None>`

Extract record changes until this date. If it is not set, we use now. If a server has ended after this date, it will be included, but the consuption reported will end on this date. If no time zone is specified, UTC will be used.

Table 7: Deprecated Variations

Group	Name
DEFAULT	extract_to

`extract_from`

> **Type** string
>
> **Default** `<None>`

Extract records that have changed after this date. This means that if a record has started before this date, and it has changed after this date (i.e. it is still running or it has ended) it will be reported. If it is not set, extract records from last run. If it is set to None and last run file is not present, it will extract records from the beginning of time. If no time zone is specified, UTC will be used.

Table 8: Deprecated Variations

Group	Name
DEFAULT	extract_from

`extractor`

> **Type** string
>
> **Default** `nova`
>
> **Valid Values** ceilometer, nova

Which extractor to use for getting the data. If you do not specify anything, nova will be used.

---

caso: keystone_auth

auth_type

> **Type** unknown type
>
> **Default** `<None>`

Authentication type to load

Table 9: Deprecated Variations

Group	Name
keystone_auth	auth_plugin

auth_section

> **Type** unknown type
>
> **Default** `<None>`

Config Section from which to load plugin specific options

cafile

> **Type** string
>
> **Default** `<None>`

PEM encoded Certificate Authority to use when verifying HTTPs connections.

certfile

> **Type** string
>
> **Default** `<None>`

PEM encoded client certificate cert file

keyfile

> **Type** string
>
> **Default** `<None>`

PEM encoded client certificate key file

insecure

> **Type** boolean
>
> **Default** `false`

Verify HTTPS connections.

timeout

> **Type** integer
>
> **Default** `<None>`

Timeout value for http requests

collect_timing

> **Type** boolean
>
> **Default** `false`

Collect per-API call timing information.

split_loggers

>   **Type** boolean

>   **Default** `false`

Log requests to multiple loggers.

auth_url

>   **Type** unknown type

>   **Default** `<None>`

Authentication URL

system_scope

>   **Type** unknown type

>   **Default** `<None>`

Scope for system operations

domain_id

>   **Type** unknown type

>   **Default** `<None>`

Domain ID to scope to

domain_name

>   **Type** unknown type

>   **Default** `<None>`

Domain name to scope to

project_id

>   **Type** unknown type

>   **Default** `<None>`

Project ID to scope to

Table 10: Deprecated Variations

Group	Name
keystone_auth	tenant-id
keystone_auth	tenant_id

project_name

>   **Type** unknown type

>   **Default** `<None>`

Project name to scope to

Table 11: Deprecated Variations

Group	Name
keystone_auth	tenant-name
keystone_auth	tenant_name

project_domain_id

> **Type** unknown type
>
> **Default** `<None>`

Domain ID containing project

project_domain_name

> **Type** unknown type
>
> **Default** `<None>`

Domain name containing project

trust_id

> **Type** unknown type
>
> **Default** `<None>`

Trust ID

default_domain_id

> **Type** unknown type
>
> **Default** `<None>`

Optional domain ID to use with v3 and v2 parameters. It will be used for both the user and project domain in v3 and ignored in v2 authentication.

default_domain_name

> **Type** unknown type
>
> **Default** `<None>`

Optional domain name to use with v3 API and v2 parameters. It will be used for both the user and project domain in v3 and ignored in v2 authentication.

user_id

> **Type** unknown type
>
> **Default** `<None>`

User id

username

> **Type** unknown type
>
> **Default** `<None>`

Username

Table 12: Deprecated Variations

Group	Name
keystone_auth	user-name
keystone_auth	user_name

user_domain_id

>> **Type** unknown type

>> **Default** `<None>`

> User's domain id

user_domain_name

>> **Type** unknown type

>> **Default** `<None>`

> User's domain name

password

>> **Type** unknown type

>> **Default** `<None>`

> User's password

caso: logstash

host

>> **Type** string

>> **Default** `localhost`

> Logstash host to send records to.

port

>> **Type** integer

>> **Default** `5000`

> Logstash server port.

caso: ssm

output_path

>> **Type** string

>> **Default** `/var/spool/apel/outgoing/openstack`

> Directory to put the generated SSM records.

max_size

>> **Type** integer

>> **Default** `100`

Maximum number of records to send per message

Usage

3.1 command line

cASO provides the `caso-extract` command to generate new records from your OpenStack deployment. `caso-extract -h` will show a complete list of available arguments.

Use the `--extract-from` argument to specify the date from when the records should be extracted. If no value is set, then cASO will extract the records from the last run. If equal to "None", then extract records from the beggining of time. If not time zone is specified, UTC will be used.

Important: If you are running an OpenStack Nova version lower than Kilo there is a bug in its API, making impossible to paginate over deleted results.

Since nova is limiting the results to 1000 by default, if you are expecting more than 1000 results you will get just the last 1000. This is important if you are publishing data for the first time, or if you are republishing all your accounting). If this is your case, adjust the `osapi_max_limit` to a larger value in `/etc/nova/nova.conf`.

3.1.1 Available options

Apart from other options, the following ones are the ones that specify how to extract accountig records:

--config-dir DIR
    Path to a config directory to pull *.conf* files from. This file set is sorted, so as to provide a predictable parse order if individual options are over-ridden. The set is parsed after the file(s) specified via previous –config-file, arguments hence over-ridden options in the directory take precedence. This option must be set from the command-line.

--config-file PATH
    Path to a config file to use. Multiple config files can be specified, with values in later files taking precedence. Defaults to None. This option must be set from the command-line.

--debug, -d

> **If set to true, the logging level will be set to DEBUG** instead of the default INFO level.

--dry-run, --dry_run
> Extract records but do not push records to SSM. This will not update the last run date.

--extract-from EXTRACT_FROM, **--extract_from** EXTRACT_FROM
> Extract records that have changed after this date. This means that if a record has started before this date, and it has changed after this date (i.e. it is still running or it has ended) it will be reported. If it is not set, extract records from last run. If it is set to None and last run file is not present, it will extract records from the beginning of time. If no time zone is specified, UTC will be used.

--extract-to EXTRACT_TO, **--extract_to** EXTRACT_TO
> Extract record changes until this date. If it is not set, we use now. If a server has ended after this date, it will be included, but the consuption reported will end on this date. If no time zone is specified, UTC will be used.

--extractor EXTRACTOR
> Which extractor to use for getting the data. If you do not specify anything, nova will be used. Allowed values: nova, ceilometer

--projects PROJECTS, **--tenants** PROJECTS
> List of projects to extract accounting records from.

3.2 Running as a cron job

The best way of running cASO is via a cron job like the following:

```
10 * * * * caso-extract
```

Symbols

C