GWOSC

Release 0.8.3.dev2+ge3636798d

Duncan Macleod

May 26, 2026

CONTENTS

1 Installation 3

2 Documentation 5

3 Index 25

Python Module Index 27

Index 29

A python interface to the Gravitational-Wave Open Science Center archive.

CHAPTER

ONE

INSTALLATION

`gwosc` can be installed via pip:

```
python -m pip install gwosc
```

or Conda:

```
conda install -c conda-forge gwosc
```

CHAPTER

TWO

DOCUMENTATION

2.1 Querying dataset information

gwosc.datasets includes functions to query for available datasets.

To search for all available datasets:

```
>>> from gwosc import datasets
>>> datasets.find_datasets()
['GW150914', 'GW151226', 'GW170104', 'GW170608', 'GW170814', 'GW170817', 'LVT151012', 'O1
↪', 'S5', 'S6']
>>> datasets.find_datasets(detector="V1")
['GW170814', 'GW170817']
>>> datasets.find_datasets(type="run")
['O1', 'S5', 'S6']
```

To query for the GPS time of an event dataset (or vice-versa):

```
>>> datasets.event_gps("GW170817")
1187008882.43
>>> datasets.event_at_gps(1187008882)
'GW170817'
```

Similar queries are available for observing run datasets:

```
>>> datasets.run_segment("O1")
(1126051217, 1137254417)
>>> datasets.run_at_gps(1135136350)  # event_gps('GW151226')
'O1'
```

To run an event query filtered by merger parameters:

```
>>> from gwosc.datasets import query_events
>>> query_events(select=["mass-1-source <= 3.0"])
['GW170817-v3', 'GW190425-v1', 'GW190425-v2', 'GW190425_081805-v3']
```

Functions

dataset_type(dataset[, host, session, pagesize])	Returns the type of the named dataset
event_at_gps(gps[, host, tol, session, pagesize])	Returns the name of the open-data event matching the GPS time

continues on next page

Table 1 – continued from previous page

event_detectors(event[, catalog, version, ...])	Returns the `set` of detectors that observed an event
event_gps(event[, catalog, version, host, ...])	Returns the GPS time of an open-data event
event_segment(event[, detector, catalog, ...])	Returns the GPS `[start, stop)` interval covered by an event dataset
find_datasets([detector, type, segment, ...])	Find datasets available on the given GW open science host
query_events(select[, host, session, pagesize])	Return a list of events filtered by the parameters in `select`
run_at_gps(gps[, host, session, pagesize])	Returns the name of the open-data run dataset matching the GPS time
run_segment(run[, host, session])	Returns the GPS `[start, stop)` interval covered by a run dataset

2.1.1 dataset_type

gwosc.datasets.**dataset_type**(dataset, host=*'https://gwosc.org'*, session=***None***, pagesize=***None***)

Returns the type of the named dataset

Parameters

dataset
The name of the dataset to match

host='https://gwosc.org'
the URL of the GWOSC host to query

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
type – the type of the dataset, one of 'run', 'event', or 'catalog'

Return type
`str`

Raises
`ValueError` – if this dataset cannot be matched

Examples

```
>>> from gwosc.datasets import dataset_type
>>> dataset_type("O1")
'run'
```

2.1.2 event_at_gps

gwosc.datasets.**event_at_gps**(gps, host=*'https://gwosc.org'*, tol=*1*, session=***None***, pagesize=***None***)

Returns the name of the open-data event matching the GPS time

This function will return the first event for which `|eventgps - gps| <= tol`.

Parameters

gps
The GPS time to locate

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

tol=1
the search window (in seconds), default: 1

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
event – the name of the matched event

Return type
str

Raises
ValueError – if no events are found matching the GPS time

Examples

```
>>> from gwosc.datasets import event_at_gps
>>> event_at_gps(1187008882)
'GW170817'
>>> event_at_gps(1187008882, tol=0.1)
ValueError: no event found within 0.1 seconds of 1187008882
```

2.1.3 event_detectors

gwosc.datasets.**event_detectors**(event, catalog=*None*, version=*None*, host='https://gwosc.org', session=*None*)

Returns the set of detectors that observed an event

Parameters

event
the name of the event to query

version=None
the version of the data release to use, defaults to the highest available version

catalog=None
name of catalogue that hosts this event

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

Returns
detectors – the set of detectors for which data file URLs are included in the data release

Return type
set

Examples

```
>>> from gwosc.datasets import event_detectors
>>> event_detectors("GW150914")
{'H1', 'L1'}
```

2.1.4 event_gps

gwosc.datasets.**event_gps**(event, catalog=*None*, version=*None*, host='https://gwosc.org', session=*None*)

Returns the GPS time of an open-data event

Parameters

event
the name of the event to query

catalog=None
name of catalogue that hosts this event

version=None
the version of the data release to use, defaults to the highest available version

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

Returns
gps – the GPS time of this event

Return type
float

Examples

```
>>> from gwosc.datasets import event_gps
>>> event_gps("GW170817")
1187008882.43
>>> event_gps("GW123456")
ValueError: no event dataset found for 'GW123456'
```

2.1.5 event_segment

gwosc.datasets.**event_segment**(event, detector=*None*, catalog=*None*, version=*None*, host='https://gwosc.org', session=*None*, pagesize=*None*)

Returns the GPS `[start, stop)` interval covered by an event dataset

Parameters

event
the name of the event

detector=None
prefix of GW detector

catalog=None
name of catalogue that hosts this event

version=None
the version of the data release to use, defaults to the highest available version

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
start, end – the GPS `[start, end)` interval covered by this event dataset

Return type
`int`

Examples

```
>>> from gwosc.datasets import event_segment
>>> event_segment("GW150914")
segment(1126257415, 1126261511)
```

2.1.6 find_datasets

`gwosc.datasets.`**find_datasets**(detector=*None*, type=*None*, segment=*None*, match=*None*, catalog=*None*, version=*None*, host='https://gwosc.org', session=*None*, pagesize=*None*)

Find datasets available on the given GW open science host

Parameters

detector=None
prefix of GW detector

type=None
type of datasets to restrict, one of `'run'`, `'event'`, or `'catalog'`

segment=None
a GPS `[start, stop)` interval to restrict matches to; datasets will match if they overlap at any point; this is not used to filter catalogs

match=None
regular expression string against which to match datasets

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
the session to use for HTTP requests

pagesize=None
the number of results per page.

Returns
datasets – the names of all matched datasets, possibly empty

Return type
`list` of `str`

Examples

(Correct as of 2018-03-14)

```
>>> from gwosc.datasets import find_datasets
>>> find_datasets()
['GW150914', 'GW151226', 'GW170104', 'GW170608', 'GW170814', 'GW170817',
 'LVT151012', 'O1', 'S5', 'S6']
>>> find_datasets(detector="V1")
['GW170814', 'GW170817']
>>> find_datasets(type="event")
['GW150914', 'GW151226', 'GW170104', 'GW170608', 'GW170814', 'GW170817',
 'LVT151012']
```

2.1.7 query_events

gwosc.datasets.**query_events**(select, host='https://gwosc.org', session=*None*, pagesize=*None*)

Return a list of events filtered by the parameters in `select`

Parameters

select
A list of strings where each element is a range constrain on the event parameters. All ranges should have inclusive ends (<= and => operators).

host='https://gwosc.org'
the URL of the GWOSC host to query

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Examples

```
>>> from gwosc.datasets import query_events
>>> query_events(
...     select=[
...         "mass-1-source >= 1.4",
...         "200 >= luminosity-distance >= 100",
...     ]
... )
['GW190425-v1', 'GW190425-v2', 'GW190425_081805-v3']
```

Notes

Operators:

- <= (or =<)
- => (or >=)

Parameters:

- `gps-time` [s],
- `mass-1-source` [solar mass],

- `mass-2-source` [solar mass],
- `network-matched-filter-snr`,
- `luminosity-distance` [Mpc],
- `chi-eff`,
- `total-mass-source` [solar mass],
- `chirp-mass` [solar mass],
- `chirp-mass-source` [solar mass],
- `redshift`,
- `far` [events/year],
- `p-astro`,
- `final-mass-source` [solar mass],

For a full description of all parameters see https://www.gwosc.org/apidocs/#event5

2.1.8 run_at_gps

`gwosc.datasets.run_at_gps`(gps, host='https://gwosc.org', session=*None*, pagesize=*None*)

Returns the name of the open-data run dataset matching the GPS time

This function will return the first event for which `start <= gps < end`

Parameters

gps
The GPS time to locate

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
run – the name of the matched observing run

Return type
`str`

Raises
`ValueError` – if no datasets are found matching the GPS time

Examples

```
>>> from gwosc.datasets import run_at_gps
>>> run_at_gps(1135136350)
'O1'
>>> run_at_gps(0)
ValueError: no run dataset found containing GPS 0
```

2.1.9 run_segment

gwosc.datasets.**run_segment**(run, host='*https://gwosc.org*', session=***None***)

Returns the GPS [start, stop) interval covered by a run dataset

Parameters

run
the name of the run dataset to query

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

Returns
start, end – the GPS [start, end) interval covered by this run dataset

Return type
int

Examples

```
>>> from gwosc.datasets import run_segment
>>> run_segment("O1")
segment(1126051217, 1137254417)
>>> run_segment("Oh dear")
ValueError: Run 'Oh dear' not found.
```

2.2 Querying for data file URLs

gwosc.locate provides functions to determine the file URLs containing data for a specific dataset.

You can search for remote data URLS based on the event name:

```
>>> from gwosc.locate import get_event_urls
>>> get_event_urls("GW150914")
['https://gwosc.org/eventapi/json/GWTC-1-confident/GW150914/v3/H-H1_GWOSC_4KHZ_R1-
↪1126259447-32.hdf5',
 'https://gwosc.org/eventapi/json/GWTC-1-confident/GW150914/v3/H-H1_GWOSC_4KHZ_R1-
↪1126257415-4096.hdf5',
 'https://gwosc.org/eventapi/json/GWTC-1-confident/GW150914/v3/L-L1_GWOSC_4KHZ_R1-
↪1126259447-32.hdf5',
 'https://gwosc.org/eventapi/json/GWTC-1-confident/GW150914/v3/L-L1_GWOSC_4KHZ_R1-
↪1126257415-4096.hdf5']
```

You can down-select the URLs using keyword arguments:

```
>>> get_event_urls("GW150914", detector="L1", duration=32)
['https://gwosc.org/eventapi/json/GWTC-1-confident/GW150914/v3/L-L1_GWOSC_4KHZ_R1-
↪1126259447-32.hdf5']
```

You can search for remote data URLs based on the GPS time interval as follows:

```
>>> from gwosc.locate import get_urls
>>> get_urls("L1", 968650000, 968660000)
['https://gwosc.org/archive/data/S6/967835648/L-L1_LOSC_4_V1-968646656-4096.hdf5',
 'https://gwosc.org/archive/data/S6/967835648/L-L1_LOSC_4_V1-968650752-4096.hdf5',
 'https://gwosc.org/archive/data/S6/967835648/L-L1_LOSC_4_V1-968654848-4096.hdf5',
 'https://gwosc.org/archive/data/S6/967835648/L-L1_LOSC_4_V1-968658944-4096.hdf5']
```

By default, this method will return the paths to HDF5 files for the 4 kHz sample-rate data, these can be specified as keyword arguments. For full information, see *get_urls()*.

Functions

get_urls(detector, start, end[, dataset, ...])	Fetch the URLs from GWOSC regarding a given GPS interval
get_run_urls(run, detector, start, end[, ...])	Fetch the URLs from GWOSC regarding a given event
get_event_urls(event[, catalog, version, ...])	Fetch the URLs from GWOSC regarding a given event

2.2.1 get_urls

gwosc.locate.**get_urls**(detector, start, end, dataset=***None***, version=***None***, sample_rate=*4096*, format=*'hdf5'*, host=*'https://gwosc.org'*, session=***None***, pagesize=***None***)

Fetch the URLs from GWOSC regarding a given GPS interval

Parameters

detector
the prefix of the relevant GW detector

start
the GPS start time of your query

end
the GPS end time of your query

dataset=None
the name of the dataset to query, e.g. `'GW150914'`

version=None
the data-release version for the selected datasets

sample_rate=4096
the sampling rate (Hz) of files you want to find

format='hdf5'
the file format (extension) you want to find

host='https://gwosc.org'
the URL of the remote GWOSC server

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
urls – the list of remote file URLs that contain data matching the relevant parameters

Return type
list of str

2.2.2 get_run_urls

gwosc.locate.**get_run_urls**(run, detector, start, end, format='hdf5', sample_rate=4096, host='https://gwosc.org', session=None, pagesize=None, **match)

Fetch the URLs from GWOSC regarding a given event

Parameters

run
the ID of the run

detector
the detector for files you want to find

start
the GPS start time of your query

end
the GPS end time of your query

format='hdf5'
the file format (extension) you want to find

sample_rate=4096
the sampling rate (Hz) of files you want to find

host='https://gwosc.org'
the URL of the remote GWOSC server

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
urls – the list of remote file URLs that contain data matching the relevant parameters

Return type
list of str

2.2.3 get_event_urls

gwosc.locate.**get_event_urls**(event, catalog=None, version=None, detector=None, start=None, end=None, format='hdf5', sample_rate=4096, host='https://gwosc.org', session=None, pagesize=None, **match)

Fetch the URLs from GWOSC regarding a given event

Parameters

event
the ID of the event

detector=None
the detector for files you want to find

format='hdf5'
the file format (extension) you want to find

sample_rate=4096
the sampling rate (Hz) of files you want to find

host='https://gwosc.org'
the URL of the remote GWOSC server

start=None
the GPS start time of your query

end=None
the GPS end time of your query

version=None
the data-release version for the selected datasets

catalog=None
the name of the catalog for the selected datasets

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
urls – the list of remote file URLs that contain data matching the relevant parameters

Return type
list of str

2.3 Querying for Timeline segments

gwosc.timeline provides functions to find segments for a given dataset.

You can search for Timeline segments, based on a flag name, and a GPS time interval as follows:

```
>>> from gwosc.timeline import get_segments
>>> get_segments("H1_DATA", 1126051217, 1126151217)
[(1126073529, 1126114861), (1126121462, 1126123267), (1126123553, 1126126832),␣
↪(1126139205, 1126139266), (1126149058, 1126151217)]
```

The output is a `list` of `[start, end)` 2-tuples which each represent a semi-open time interval.

For documentation on what flags are available, for example for the O1 science run, see the O1 data release page (*Data Quality*).

Functions

DEFAULT_URL	str(object='') -> str str(bytes_or_buffer[, encoding[, errors]]) -> str
get_segments(flag, start, end[, host, ...])	Return the [start, end) GPS segments for this flag
timeline_url(flag, start, end[, host, session])	Returns the Timeline JSON URL for a flag name and GPS interval

2.3.1 DEFAULT_URL

gwosc.timeline.**DEFAULT_URL** = 'https://gwosc.org'

str(object='') -> str str(bytes_or_buffer[, encoding[, errors]]) -> str

Create a new string object from the given object. If encoding or errors is specified, then the object must expose a data buffer that will be decoded using the given encoding and error handler. Otherwise, returns the result of object.__str__() (if defined) or repr(object). encoding defaults to sys.getdefaultencoding(). errors defaults to 'strict'.

2.3.2 get_segments

gwosc.timeline.**get_segments**(flag, start, end, host='https://gwosc.org', session=None, pagesize=None)

Return the [start, end) GPS segments for this flag

Parameters

flag
name of flag, e.g. 'H1_DATA'

start
the GPS start time of your query

end
the GPS end time of your query

host='https://gwosc.org'
the URL of the remote GWOSC server

session=None
HTTP session to use for making requests, defaults to using a new session for each API call

pagesize=None
the number of results per page

Returns
segments – a list of [a, b) GPS segments

Return type
list of (int, int) tuples

2.3.3 timeline_url

gwosc.timeline.**timeline_url**(flag, start, end, host='https://gwosc.org', session=None)

Returns the Timeline JSON URL for a flag name and GPS interval

2.4 Low-level API

The *gwosc.api* package provides low-level interface functions that handle direct requests to the GWOSC host. Inside the *gwosc.api* package there are two modules, each one dealing with the 2 available versions of the API.

- *gwosc.api.v1*: Legacy API functions for backward compatibility
- *gwosc.api.v2*: Newer API functions with improved functionality

Given the sharp increase of detections in recent runs, we rebuilt our public API to cope with increased response length, and high-frequency requests. API v2 introduces pagination and throttling features for all resources to guarantee scalability, and a smoother experience for developers and end users. To learn more, visit https://gwosc.org/api/.

Backward Compatibility

For backward compatibility, the main *gwosc.api* module imports all functions from the v1 API. This means existing code that imports from *gwosc.api* will continue to work without changes. *gwosc.api* provides the low-level interface functions that handle direct requests to the GWOSC host.

gwosc.api.**DEFAULT_URL** = 'https://gwosc.org'

The default GWOSC host URL

API v2 (Current)

gwosc.api.v2.**fetch_allowed_params**(host='https://gwosc.org', session=None)

Return a list with the "default parameters".

These parameters are almost always estimated for event detections. Use the parameter names (strings) to filter events in the *fetch_event_versions()* `select` argument.

Parameters

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

Returns
params – A list of parameter names.

Return type
`list[str]`

gwosc.api.v2.**fetch_catalogs**(host='https://gwosc.org', session=None, pagesize=None)

Returns a list with all catalogs.

Parameters

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – A list of catalogs.

Return type
`iterable[dict]`

gwosc.api.v2.**fetch_event_strain_data**(event, detector=None, version=None, catalog=None, sample_rate=None, duration=None, format='hdf5', host='https://gwosc.org', session=None, pagesize=None)

Return strain file objects from single-event releases.

Parameters

event
the ID of an event, e.g. `'GW150914'`.

detector=None
the prefix of the GW detector, e.g. `'L1'`.

version=None
the version number of the requested event.

catalog=None
the catalog in which the requested event appears.

sample_rate=None
the sample rate of the strain file data, either 4096 or 16384 [Hz].

duration=None
the duration of the strain file, 32 or 4096 [s].

format='hdf5'
the file format of the strain file. One of `'hdf'`, `'gwf'`, `'txt'`.

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – An iterable of strain file dictionaries.

Return type
`iterable[dict]`

`gwosc.api.v2.fetch_event_version(event, catalog=None, version=None, host='https://gwosc.org', session=None)`

Returns an event.

Parameters

event
the name of the event.

catalog=None
name of catalogue that hosts this event.

version=None
the version of the data release to use, defaults to the highest available version.

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

Returns
data – A dictionary with the event detail.

Return type
`dict`

`gwosc.api.v2.fetch_event_versions(name=None, segment=None, catalogs=None, select=None, host='https://gwosc.org', session=None, pagesize=None)`

Returns an event.

Parameters

name=None
a full or partial name for an event.

segment=None
a gps time tuple (start, end) to restrict the search.

catalogs=None
a single catalog name or a list of catalog names.

select=None
a dictionary with query parameters, e.g. {'min-p-astro': 0.5}.

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – A list of event version dictionaries.

Return type
`iterable[dict]`

`gwosc.api.v2.fetch_json(url, session=None, **kwargs)`

Fetch JSON data from a remote URL.

Parameters

url
the remote URL to fetch

session=None
the session to use for the request. If None, falls back to direct requests.get()

****kwargs**
other keyword arguments are passed directly to `requests.get()` or `requests.Session.get()`

Returns
data – the data fetched from `url` as parsed by `requests.Response.json()`

Return type
`dict` or `list`

See also

`json.loads`
for details of the JSON parsing

`gwosc.api.v2.fetch_run(run, host='https://gwosc.org', session=None)`

Return a run detail.

Parameters

run
the name of the run, e.g. O1.

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

Returns
data – A dictionary with the run detail.

Return type
dict

gwosc.api.v2.**fetch_run_strain_files**(run=*None*, detector=*None*, start=*None*, end=*None*, sample_rate=*None*, host='https://gwosc.org', session=*None*, pagesize=*None*)

Return strain file objects from bulk-data releases.

Parameters

run=None
the ID of a run, e.g. `'O1'`.

detector=None
the prefix of the GW detector, e.g. `'L1'`.

start=None
the GPS start of the desired interval.

end=None
the GPS end of the desired interval.

sample_rate=None
the sample rate of the strain file data, either 4096 or 16384 [Hz].

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – An iterable of strain file dictionaries.

Return type
iterable[dict]

gwosc.api.v2.**fetch_runs**(host='https://gwosc.org', session=*None*, pagesize=*None*)

Return a list of all past runs.

Parameters

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – An iterable of runs.

Return type
`iterable[dict]`

`gwosc.api.v2.fetch_segments(flag, start, end, host='https://gwosc.org', session=None, pagesize=None)`

Return segment dictionaries in the (start, end) GPS time interval.

Parameters

flag
name of flag, e.g. `'H1_DATA'`.

start
the GPS start time of your query.

end
the GPS end time of your query.

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org.

session=None
the session to use for HTTP requests.

pagesize=None
the number of results per page.

Returns
data – An iterable of segment dictionaries.

Return type
`iterable[dict]`

`gwosc.api.v2.produce_fetched_objects(url, session=None)`

API v1 (Legacy)

`gwosc.api.v1.fetch_allevents_json(full=False, host='https://gwosc.org')`

Returns the JSON metadata for the allevents API

Parameters

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returned by `requests.Response.json()`

Return type
`dict` or `list`

`gwosc.api.v1.fetch_allowed_params_json(host='https://gwosc.org')`

`gwosc.api.v1.fetch_catalog_json(catalog, host='https://gwosc.org')`

Returns the JSON metadata for the given catalogue

Parameters

catalog
the name of the event catalog, e.g. `GWTC-1-confident`

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returnend by `requests.Response.json()`

Return type
`dict` or `list`

`gwosc.api.v1.fetch_cataloglist_json(host='https://gwosc.org')`

Returns the JSON metadata for the catalogue list.

Parameters

host='https://gwosc.org'
the URL of the GWOSC host to query

Returns
data – the JSON data retrieved from GWOSC and returned by `requests.Response.json()`

Return type
`dict` or `list`

`gwosc.api.v1.fetch_dataset_json(gpsstart, gpsend, host='https://gwosc.org')`

Returns the JSON metadata for all datasets matching the GPS interval

Parameters

gpsstart
the GPS start of the desired interval

gpsend
the GPS end of the desired interval

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returned by `json.loads`

Return type
`dict` or `list`

`gwosc.api.v1.fetch_event_json(event, catalog=None, version=None, host='https://gwosc.org')`

Returns the JSON metadata for the given event.

By default, this function queries across all catalogs and all data-release versions, returning the highest available version, unless the `version` and/or `catalog` keywords are specified.

Parameters

event
the name of the event to query

catalog=None
name of catalogue that hosts this event

version=None
restrict query to a given data-release version

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returned by `json.loads`

Return type
`dict` or `list`

`gwosc.api.v1.fetch_filtered_events_json(select, host='https://gwosc.org')`

Return the JSON metadata for the events constrained by select

Parameters

select
a list of range constrains for the events. All ranges should have inclusive ends (<= and >= operators).

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returnend by `requests.Response.json()`

Return type
`dict` or `list`

Example

```
>>> fetch_filtered_events_json(
...     select=[
...         "mass-1-source <= 5",
...         "mass-2-source =< 10",
...         "10 <= luminosity-distance <= 100",
...     ]
... )
```

`gwosc.api.v1.fetch_json(url, **kwargs)`

Fetch JSON data from a remote URL

Parameters

url
the remote URL to fetch

kwargs : dict
other keyword arguments are passed directly to `requests.get()`

Returns
data – the data fetched from `url` as parsed by `requests.Response.json()`

Return type
`dict` or `list`

See also

`json.loads`
for details of the JSON parsing

gwosc.api.v1.**fetch_run_json**(run, detector, gpsstart=0, gpsend=*99999999999*, host='*https://gwosc.org*')

Returns the JSON metadata for the given science run parameters

Parameters

run
the name of the science run, e.g. `'O1'`

detector
the prefix of the GW detector, e.g. `'L1'`

gpsstart=0
the GPS start of the desired interval

gpsend=99999999999
the GPS end of the desired interval

host='https://gwosc.org'
the URL of the GWOSC host to query, defaults to https://gwosc.org

Returns
data – the JSON data retrieved from GWOSC and returned by `json.loads`

Return type
`dict` or `list`

CHAPTER

THREE

INDEX

- genindex
- modindex

PYTHON MODULE INDEX

g

gwosc.api, 17
gwosc.api.v1, 21
gwosc.api.v2, 17
gwosc.datasets, 5
gwosc.locate, 12
gwosc.timeline, 15

INDEX

D

dataset_type() (*in module gwosc.datasets*), 6
DEFAULT_URL (*in module gwosc.api*), 17
DEFAULT_URL (*in module gwosc.timeline*), 16

E

event_at_gps() (*in module gwosc.datasets*), 6
event_detectors() (*in module gwosc.datasets*), 7
event_gps() (*in module gwosc.datasets*), 8
event_segment() (*in module gwosc.datasets*), 8

F

fetch_allevents_json() (*in module gwosc.api.v1*), 21
fetch_allowed_params() (*in module gwosc.api.v2*), 17
fetch_allowed_params_json() (*in module gwosc.api.v1*), 21
fetch_catalog_json() (*in module gwosc.api.v1*), 21
fetch_cataloglist_json() (*in module gwosc.api.v1*), 22
fetch_catalogs() (*in module gwosc.api.v2*), 17
fetch_dataset_json() (*in module gwosc.api.v1*), 22
fetch_event_json() (*in module gwosc.api.v1*), 22
fetch_event_strain_data() (*in module gwosc.api.v2*), 17
fetch_event_version() (*in module gwosc.api.v2*), 18
fetch_event_versions() (*in module gwosc.api.v2*), 18
fetch_filtered_events_json() (*in module gwosc.api.v1*), 23
fetch_json() (*in module gwosc.api.v1*), 23
fetch_json() (*in module gwosc.api.v2*), 19
fetch_run() (*in module gwosc.api.v2*), 19
fetch_run_json() (*in module gwosc.api.v1*), 23
fetch_run_strain_files() (*in module gwosc.api.v2*), 20
fetch_runs() (*in module gwosc.api.v2*), 20
fetch_segments() (*in module gwosc.api.v2*), 21
find_datasets() (*in module gwosc.datasets*), 9

G

get_event_urls() (*in module gwosc.locate*), 14
get_run_urls() (*in module gwosc.locate*), 14
get_segments() (*in module gwosc.timeline*), 16
get_urls() (*in module gwosc.locate*), 13
gwosc.api
    module, 17
gwosc.api.v1
    module, 21
gwosc.api.v2
    module, 17
gwosc.datasets
    module, 5
gwosc.locate
    module, 12
gwosc.timeline
    module, 15

M

module
    gwosc.api, 17
    gwosc.api.v1, 21
    gwosc.api.v2, 17
    gwosc.datasets, 5
    gwosc.locate, 12
    gwosc.timeline, 15

P

produce_fetched_objects() (*in module gwosc.api.v2*), 21

Q

query_events() (*in module gwosc.datasets*), 10

R

run_at_gps() (*in module gwosc.datasets*), 11
run_segment() (*in module gwosc.datasets*), 12

T

timeline_url() (*in module gwosc.timeline*), 16