decisionengine

Release 1.1.1

Fermi Research Alliance, LLC.

Jul 02, 2020

CONTENTS

The decisionengine_modules package contains provider specific implementations of base classes provided by the decision engine framework.

CHAPTER

ONE

RELEASE NOTES

1.1 Release 1.2.0

In this release:

- Switched to python3
- NERSC newt interface adapted to changed NERSC RESTful API
- Improved unit tests
- Added Jenkins CI pipeline

1.2 Patches:

30d928b : clone version 1.2.0 of decisionengine

ae7c5a6 : Jenkins pipeline improvements (#236)

310befd : T198 (#235)

a65886d : Fix import as reported in : https://github.com/HEPCloud/decisionengin… (#232)

93711cc : Run coveralls even if tests fail (#229)

03d763a : Jenkins pipeline improvements (#230)

f48d30f : Fix/223 (#228)

c8aa262 : github ticket 199 (#222)

0323bda : Address : https://github.com/HEPCloud/decisionengine_modules/issues/224 (#226)

62e4df6 : Add support to run CI on Jenkins (#221)

5ab1541 : bump master version to 1.2.0 (for now) (#219)

bc19c65 : decisionengine_modules/NERSC: Added retry loop for NERSC API Calls (#220)

41a50de : Sync up pep8speaks and run_pylint.sh with decisionengine settings (#218)

db4634f : silence pylint error (#217)

1b95141 : Fix whitespace around operator error

746ea38 : ignore W503

8a8b5f4 : remove unused variable

a6668bf : fix PEP8 warnings

13773ee : address pep8 warnings

6bea4ca : silence pylint error

f589895 : Pass sort=True parameter to fix future warning (#215)

a1d0507 : fixing pep8 warning

a10bd17 : debugging one import error

ec501ad : make coveralls.io links work

deab1a7 : T201 (#204)

69f2645 : Add coveragerc

6d8a5f5 : decisionengine_modules/NERSC: Make Nersc API call backward-compatible with old config (#196)

a7e0af9 : Only run Github Actions for python3.6 (#24)

CHAPTER

TWO

DEVELOPER DOCUMENTATION

First command `cd` is just to make sure that you end up in a directory that will contain two subdirectory `decisionengine` and `decisionengine_modules`. Of course this can be done in any directory, not necessarily home directory.

2.1 Decisionengine_modules

2.1.1 Prerequisites:

In Addition to above installed packages

```
yum install condor
pip3 install htcondor boto boto3 google_auth google-api-python-client gcs-oauth2-boto-
↪plugin
```

2.1.2 Test

```
cd

git clone https://github.com/HEPCloud/decisionengine_modules
python3 -m pytest decisionengine_modules
```

Current status:

```
[root@fermicloud371 tmp]# python3 -m pytest decisionengine_modules
==================================== test session starts␣
↪=====================================
platform linux -- Python 3.6.8, pytest-5.3.5, py-1.8.1, pluggy-0.13.1
rootdir: /root/junjk
collected 85 items

decisionengine_modules/AWS/tests/test_AWSInstancePerformance.py ..                 ␣
↪ [  2%]
decisionengine_modules/AWS/tests/test_AWSJobLimits.py ..                           ␣
↪ [  4%]
decisionengine_modules/AWS/tests/test_AWSOccupancyWithSourceProxy.py ..            ␣
↪ [  7%]
decisionengine_modules/AWS/tests/test_AWSSpotPriceWithSourceProxy.py ..            ␣
↪ [  9%]
decisionengine_modules/AWS/tests/test_AWS_figure_of_merit_publisher.py ..          ␣
↪ [ 11%]
```

(continues on next page)

(continued from previous page)

```
decisionengine_modules/AWS/tests/test_AWS_price_performance_publisher.py ..              ␣
↪ [ 14%]
decisionengine_modules/AWS/tests/test_FigureOfMerit.py ...                               ␣
↪ [ 17%]
decisionengine_modules/tests/test_AwsBurnRate.py ..                                      ␣
↪ [ 20%]
decisionengine_modules/tests/test_GCEBillingInfo.py ..                                   ␣
↪ [ 22%]
decisionengine_modules/tests/test_GCEFigureOfMerit_publisher.py ..                       ␣
↪ [ 24%]
decisionengine_modules/tests/test_GCEInstancePerformanceInfo.py ..                       ␣
↪ [ 27%]
decisionengine_modules/tests/test_GCEPricePerformance_publisher.py ..                    ␣
↪ [ 29%]
decisionengine_modules/tests/test_GCEResourceLimits.py ..                                ␣
↪ [ 31%]
decisionengine_modules/tests/test_GceBurnRate.py ..                                      ␣
↪ [ 34%]
decisionengine_modules/tests/test_GceFigureOfMerit.py ..                                 ␣
↪ [ 36%]
decisionengine_modules/tests/test_GceOccupancy.py ..                                     ␣
↪ [ 38%]
decisionengine_modules/tests/test_NerscAllocationInfo.py ..                              ␣
↪ [ 41%]
decisionengine_modules/tests/test_NerscFigureOfMerit.py ..                               ␣
↪ [ 43%]
decisionengine_modules/tests/test_NerscFigureOfMerit_publisher.py ..                     ␣
↪ [ 45%]
decisionengine_modules/tests/test_NerscInstancePerformance.py ..                         ␣
↪ [ 48%]
decisionengine_modules/tests/test_NerscJobInfo.py ..                                     ␣
↪ [ 50%]
decisionengine_modules/tests/test_factory_client.py ....                                 ␣
↪ [ 55%]
decisionengine_modules/tests/test_factory_entries.py ....                                ␣
↪ [ 60%]
decisionengine_modules/tests/test_factory_global.py ....                                 ␣
↪ [ 64%]
decisionengine_modules/tests/test_fomorderplugin.py ....                                 ␣
↪ [ 69%]
decisionengine_modules/tests/test_grid_figure_of_merit.py .                              ␣
↪ [ 70%]
decisionengine_modules/tests/test_htcondor_query.py ....                                 ␣
↪ [ 75%]
decisionengine_modules/tests/test_job_clustering.py .....                                ␣
↪ [ 81%]
decisionengine_modules/tests/test_job_clustering_publisher.py ..                         ␣
↪ [ 83%]
decisionengine_modules/tests/test_job_q.py ...                                           ␣
↪ [ 87%]
decisionengine_modules/tests/test_slots.py ..                                            ␣
↪ [ 89%]
decisionengine_modules/tests/glideinwms/publishers/test_decisionenginemonitor.py ...     ␣
↪ [ 92%]
decisionengine_modules/tests/glideinwms/publishers/test_fe_group_classads.py ...         ␣
↪ [ 96%]
decisionengine_modules/tests/glideinwms/publishers/test_glideclientglobal.py ...         ␣
↪ [100%]
```

(continues on next page)

(continued from previous page)

```
====================================== warnings summary␣
↪======================================
/usr/local/lib/python3.6/site-packages/boto/plugin.py:40
  /usr/local/lib/python3.6/site-packages/boto/plugin.py:40: DeprecationWarning: the␣
↪imp module is deprecated in favour of importlib; see the module's documentation for␣
↪alternative uses
    import imp

-- Docs: https://docs.pytest.org/en/latest/warnings.html
=============================== 85 passed, 1 warning in 9.73s␣
↪================================
```

CHAPTER

THREE

SOURCE CODE

3.1 Welcome to decisionengine_modules's documentation!

3.1.1 decisionengine_modules package

Subpackages

decisionengine_modules.AWS package

Subpackages

decisionengine_modules.AWS.publishers package

Submodules

decisionengine_modules.AWS.publishers.AWS_burn_rate module

decisionengine_modules.AWS.publishers.AWS_figure_of_merit module

decisionengine_modules.AWS.publishers.AWS_generic_publisher module

decisionengine_modules.AWS.publishers.AWS_price_performance module

Module contents

decisionengine_modules.AWS.sources package

Submodules

decisionengine_modules.AWS.sources.AWSBurnRateSourceProxy module

decisionengine_modules.AWS.sources.AWSInstancePerformance module

decisionengine_modules.AWS.sources.AWSJobLimits module

decisionengine_modules.AWS.sources.AWSOccupancy module

decisionengine_modules.AWS.sources.AWSOccupancyWithSourceProxy module

decisionengine_modules.AWS.sources.AWSSpotPrice module

decisionengine_modules.AWS.sources.AWSSpotPriceWithSourceProxy module

decisionengine_modules.AWS.sources.BillingInfo module

decisionengine_modules.AWS.sources.BillingInfoSourceProxy module

decisionengine_modules.AWS.sources.DEAccountContants module

This data is taken from billing and arranged as dictionary of AccountConstants instead of separate class for each billing class. This way is more covenient to use in programms.

class `decisionengine_modules.AWS.sources.DEAccountContants.`**`AccountConstants`**`(`*constants={}*`)`

Bases: `object`

`info``()`

`decisionengine_modules.AWS.sources.DEAccountContants.`**`load_constants`**`(`*constants_file*`)`

Load constants from file.

Parameters **`constants_file`** (`file`) – configuration file as python dict

Return type *`AccountConstants`*

decisionengine_modules.AWS.sources.FigureOfMeritSourceProxy module

decisionengine_modules.AWS.sources.FinancialParameters module

decisionengine_modules.AWS.sources.occupancy_config_sample module

decisionengine_modules.AWS.sources.spot_price_config_sample module

Module contents

decisionengine_modules.AWS.transforms package

Submodules

decisionengine_modules.AWS.transforms.AwsBurnRate module

decisionengine_modules.AWS.transforms.FigureOfMerit module

Module contents

Submodules

decisionengine_modules.AWS.AWSSpotPrice module

decisionengine_modules.AWS.CloudBudget module

decisionengine_modules.AWS.CloudRequests module

decisionengine_modules.AWS.HPCAllocation module

decisionengine_modules.AWS.HTCondorJobQ module

decisionengine_modules.AWS.LocalCapacity module

decisionengine_modules.AWS.ProvisionerResourceList module

decisionengine_modules.AWS.pandas_scratch_pad module

`decisionengine_modules.AWS.pandas_scratch_pad.`**`load_data_frame`**`(`*list_of_dicts*`)`

Module contents

decisionengine_modules.GCE package

Subpackages

decisionengine_modules.GCE.publishers package

Submodules

decisionengine_modules.GCE.publishers.GCEBurnRatePublisher module

decisionengine_modules.GCE.publishers.GCEFigureOfMerit_publisher module

decisionengine_modules.GCE.publishers.GCEPricePerformance_publisher module

Module contents

decisionengine_modules.GCE.sources package

Submodules

decisionengine_modules.GCE.sources.GCEBillingInfo module

decisionengine_modules.GCE.sources.GCEBillingInfoSourceProxy module

decisionengine_modules.GCE.sources.GCEInstancePerformance module

decisionengine_modules.GCE.sources.GCEResourceLimits module

decisionengine_modules.GCE.sources.GceBurnRateSourceProxy module

decisionengine_modules.GCE.sources.GceFigureOfMeritSourceProxy module

decisionengine_modules.GCE.sources.GceOccupancy module

Module contents

decisionengine_modules.GCE.transforms package

Submodules

decisionengine_modules.GCE.transforms.GceBurnRate module

decisionengine_modules.GCE.transforms.GceFigureOfMerit module

Module contents

Module contents

decisionengine_modules.NERSC package

Subpackages

decisionengine_modules.NERSC.publishers package

Submodules

decisionengine_modules.NERSC.publishers.NerscFigureOfMerit_publisher module

Module contents

decisionengine_modules.NERSC.sources package

Submodules

decisionengine_modules.NERSC.sources.NerscAllocationInfo module

decisionengine_modules.NERSC.sources.NerscAllocationInfoSourceProxy module

decisionengine_modules.NERSC.sources.NerscFigureOfMeritSourceProxy module

decisionengine_modules.NERSC.sources.NerscInstancePerformance module

decisionengine_modules.NERSC.sources.NerscJobInfo module

Module contents

decisionengine_modules.NERSC.transforms package

Submodules

decisionengine_modules.NERSC.transforms.CompareNerscFactoryJobs module

decisionengine_modules.NERSC.transforms.CompareNerscUserpoolSlots module

decisionengine_modules.NERSC.transforms.NerscFigureOfMerit module

Module contents

decisionengine_modules.NERSC.util package

Submodules

decisionengine_modules.NERSC.util.newt module

Newt API interface

class `decisionengine_modules.NERSC.util.newt.`**`Newt`**(*password_file*, *newt_base_url=None*, *num_retries=0*, *retry_backoff_factor=0*)

Bases: `object`

`_add_retries_to_session`()

Adds retries to requests Session for requests to NEWT URLs :return: void

`_login`()

Establishes newt Session :return: void

`get_queue`(*system*, *query=None*)

Returns information about queues controlled by optional query name parameter :param system: string system name (like 'cori', 'edison' ..) :param query: string query (optional) :return: json containing queue information

`get_status`(*system=None*)

Returns system status for a given system (if provided) or all systems :param system: string name of the system (optional) :return: json containing system status

`get_usage`(*username*)

Returns allocation and usage for the given user using NEWT IRIS API :param username: string username :return: json containing allocation and usage information for a given user

Module contents

Module contents

decisionengine_modules.graphite package

Subpackages

decisionengine_modules.graphite.publishers package

Submodules

decisionengine_modules.graphite.publishers.generic_publisher module

Module contents

Module contents

decisionengine_modules.htcondor package

Subpackages

decisionengine_modules.htcondor.publishers package

Submodules

decisionengine_modules.htcondor.publishers.publisher module

Module contents

decisionengine_modules.htcondor.sources package

Submodules

decisionengine_modules.htcondor.sources.job_q module

decisionengine_modules.htcondor.sources.job_q_source_proxy module

decisionengine_modules.htcondor.sources.slots module

decisionengine_modules.htcondor.sources.slots_source_proxy module

decisionengine_modules.htcondor.sources.source module

Module contents

Submodules

decisionengine_modules.htcondor.htcondor_query module

Module contents

decisionengine_modules.util package

Submodules

decisionengine_modules.util.figure_of_merit module

Calculate figure of merit

`decisionengine_modules.util.figure_of_merit.`**`figure_of_merit`**(*performance*, *running*, *allowed*, *idle=None*, *max_idle=None*)

decisionengine_modules.util.retry_function module

`decisionengine_modules.util.retry_function.`**`retry_on_error`**(*nretries=1*, *retry_interval=2*, *backoff=True*)

`decisionengine_modules.util.retry_function.`**`retry_wrapper`**(*f*, *nretries=1*, *retry_interval=2*, *backoff=True*)

Retry on error with parameters of how many times (nretries) and interval in seconds (retry_interval). If the function to be decorated is an instance method and the values come from a configuration, then this decorator can retrieve these two parameters from instance attributes with the same name. Otherwise, use the default values or pass new values to the decorator.

decisionengine_modules.util.test_retry_function module

`class` `decisionengine_modules.util.test_retry_function.`**`Dummy`**(*name='Dummy'*)

Bases: `object`

`_func_failure`(*foo*, *input2=3*)

`_func_success`(*foo*, *input2=3*)

`func_failure`(*foo*, *input2=3*)

`func_success`(*foo*, *input2=3*)

`decisionengine_modules.util.test_retry_function.`**`test_all`**()

decisionengine_modules.util.testutils module

Module contents

Submodules

decisionengine_modules.graphite_client module

class `decisionengine_modules.graphite_client.`**`Graphite`**(*host='fifemondata.fnal.gov'*, *pickle_port=2004*)

Bases: `object`

`send_dict`(*namespace*, *data*, *debug_print=True*, *send_data=True*)

send data contained in dictionary as {k: v} to graphite dataset $namespace.k with current timestamp

`decisionengine_modules.graphite_client.`**`sanitize_key`**(*key*)

decisionengine_modules.load_config module

`decisionengine_modules.load_config.`**`load`**(*python_file*, *retries=0*, *timeout=0*)

Load constants from file.

Parameters **`constants_file`** (`file`) – configuration file as python dict

Return type `object`

Module contents

3.2 Indices and tables

- genindex
- modindex
- search

CHAPTER FOUR

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX