# decisionengine

Release 1.1.1

Fermi Research Alliance, LLC.

# **CONTENTS**

1	Release Notes	3
2	<b>Developer Documentation</b>	7
3	Source code	11
4	Indices and tables	29
Ру	ython Module Index	31
In	dex	33

The Decision Engine is a critical component of the HEP Cloud Facility. It provides the functionality of resource scheduling for disparate resource providers, including those which may have a cost or a restricted allocation of cycles

CONTENTS 1

2 CONTENTS

**CHAPTER** 

ONE

# **RELEASE NOTES**

# 1.1 Release 1.1.0

#### In this release:

- Fixed. https://github.com/HEPCloud/decisionengine\_modules/issues/108 "Supply Postgres script to delete fields in main database before a certain date"
- significant code cleanup and pep8 compliance
- · unit test work
- CI (GitHub actions and Travis) is introduced

#### commits

f894b1d: Skip unittest (#77)

632e64b: Add ipython

f681a79: Make python 2.7 tests run on 1.1 branch

d6a32c0: implementation of data reaper (#75)

2ad8614: Use sparse checkout for first checkout to get .github/actions (#65)

812f032: Cat output of pytest log Exit pylint entrypoint with the line count of pep8 and pylint logs Deal with (detach

from ...) Only tar up (S)RPMS dirs for rpm build.

6b05ec7: Fix errors reported by run\_pylint (#62)

d9f5b66 : Setup pep8speaks

c3b8ac2: Run github actions as non-root uid. Install packages in virtualenv and remove system rpms.

ae01f9e: Support Python 3 for Boost Python

579761c: Support Python 3 for Boost Python

044b979: Remove unnecessary using declarations.

00f6d00 : Add extra header dependency due to Boost Python ommission.

24e0795 : Apply clang-format

17c17f9: Remove JSON dependency.

faa0b22: Massive cleanup.

07b555f: Updates to Github Actions to allow building with python3.6

fef6c11: Fix errors when running pylint.sh multiple times

```
da6f077: Autopep8 -i fixes
39fe5b3: TaskManager: fix calling log_exception with correct number of arguments and minor format changes to
reduce PEP8 warnings
17396da: logicengine: get rid of compuler warnings
01dc3d1: Only track what we need
b609d73 : Configure coveralls (and some minor cleanup)
bd9ed5e: Many C++ cleanups
2a61876: Add Badges
c864f27: Do not call pytest fixtures directly.
307db5f: white space fix
882b58f: fix unit tests
1da687c: Replace Boost facilities with C++ STL ones.
5a6e6b1: Run tests on push
8404245 : Add missing Boost regex library dependency.
ceb5fe7: Apply clang-format to files that were missed earlier.
3de9940 : Apply clang-format to C++ code.
8a8f560: Cache venv directory instead
ad017ce: Build private boost for testing
928c64a: Test pip cache
358939a: Adjust CMakeLists.txt files to use correct Python versions
9f0ddb3: Add pylint github action.
5e6ce4a: Remove more unused C++ files.
63717fe: Setup travis to use new cmake var
74fab2a: Use cmake argumement -DPYVER=3.6 to build python3 library https://fermicloud140.fnal.gov/reviews/r/
843f30c: Minor cleanups per travis-lint
a538cac: Remove unused C++ files.
4c9d125: Update repo where action is taken from
87fb2d9: Update rpms installed in docker image. Update entrypoint.sh to use cmake3.
199ee87: Find python3 libraries using cmake3 from epel rpm Also need to install python3-devel
4c79d2c: Remove unnused GNUmakefiles.
94342ee: Add unit test as a Github Action
1a0e102: more advanced travis.yml
0be413f: Add helper file for pip
7794327: Make recursive import happy
7005c78 : Add simple target
```

de8b0fa: python3 compliance: replace string.join() where appropriate, handle UserDict

2662e6c: note required packages

3b87119 : Add missing header includes.

3e79b84 : Remove defunct code and its tests

b1dbe1a: Ensure attribs are defined at init

c4ad78a: Correct logger arguments do avoid duplicate string parse

a8dcc67: Remove unused imports (per pylint)

d3502b5: Remove obsolete CVS directories.

d744111: add six module to the list of required modules

0a9b1e8: Fix class declaration

b83157e: Handle metaclasses

549f33b: Add config for Travis CI

ee71044: Drop trailing white space

3f82af6: Python3 forward compatible syntax

28bf291 : Add safe (for python 2.7) python3 compatible syntax

1d1d76f : prepare for python3

1.1. Release 1.1.0 5

**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 framework

# 2.1.1 Prerequisites:

```
yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
yum install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/
-pgdg-redhat-repo-latest.noarch.rpm
yum install -y python3 python3-pip cmake3 boost-devel python36-devel boost-python36-
-devel postgresql11 postgresql11-server
pip3 install pandas DBUtils psycopg2-binary tabulate mock pytest
```

#### 2.1.2 Build & test

```
cd
git clone https://github.com/HEPCloud/decisionengine
export PYTHONPATH='pwd'
mkdir decisionengine/framework/logicengine/cxx/build
cd decisionengine/framework/logicengine/cxx/build
cmake3 .. -DPYVER=3.6
make -j <number> # say number of CPUs on your box
cd ../../
ln -s cxx/build/ErrorHandler/RE.so
ln -s cxx/build/ErrorHandler/libLogicEngine.so
export LD_LIBRARY_PATH=`pwd`
cd ../../
#pytest -v --tb=native
python3 -m pytest
   ======= test session starts...
→=========
platform linux -- Python 3.6.8, pytest-5.3.5, py-1.8.1, pluggy-0.13.1
rootdir: /root/junjk/decisionengine
collected 26 items
```

(continues on next page)

(continued from previous page)

# 2.2 Decisionengine\_modules

# 2.2.1 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
```

#### 2.2.2 Test

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

#### Current status:

(continued from previous page)

```
decisionengine_modules/AWS/tests/test_AWSSpotPriceWithSourceProxy.py ..
→ [ 9%]
decisionengine_modules/AWS/tests/test_AWS_figure_of_merit_publisher.py ..
decisionengine_modules/AWS/tests/test_AWS_price_performance_publisher.py ...
decisionengine_modules/AWS/tests/test_FigureOfMerit.py ...

→ [ 17%]

decisionengine_modules/tests/test_AwsBurnRate.py ..

→ [ 20%]

decisionengine_modules/tests/test_GCEBillingInfo.py ..
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%]

{\tt decisionengine\_modules/tests/test\_GceFigureOfMerit.py} \ \dots
→ [ 36%]
decisionengine_modules/tests/test_GceOccupancy.py ...
decisionengine_modules/tests/test_NerscAllocationInfo.py ..
decisionengine_modules/tests/test_NerscFigureOfMerit.py ..
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 ..
decisionengine_modules/tests/test_job_q.py ...
→ [ 87%]
decisionengine_modules/tests/test_slots.py ..
decisionengine_modules/tests/glideinwms/publishers/test_decisionenginemonitor.py ...
                                                                          (continues on next page)
→ [ 92%]
```

(continued from previous page)

**CHAPTER** 

THREE

# **SOURCE CODE**

# 3.1 Welcome to decisionengine's documentation!

# 3.1.1 decisionengine package

**Subpackages** 

decisionengine.framework package

**Subpackages** 

decisionengine.framework.configmanager package

**Submodules** 

decisionengine.framework.configmanager.ConfigManager module

```
class decisionengine.framework.configmanager.ConfigManager.Bases: object
    check_keys (channel_conf_dict)
        check that channel config has mandatory keys :type data: dict
    static create (module_name, class_name, parameters)
    get_channels()
    get_global_config()
    get_produces (channel_config)
    is_updated()
    load()
    reload()
    validate_channel (channel)
        Validate channels :type channel: dict
```

#### **Module contents**

decisionengine.framework.dataspace package

#### Subpackages

decisionengine.framework.dataspace.datasources package

#### **Submodules**

decisionengine.framework.dataspace.datasources.postgresql module

```
class decisionengine.framework.dataspace.datasources.postgresql.Postgresql (config_dict)
     Bases: decisionengine.framework.dataspace.datasource.DataSource
     Implementation of postgresql data source
     _Postgresql__query (query_string, values=None, cursor_factory=None)
     _abc_cache = <_weakrefset.WeakSet object>
     _abc_negative_cache = <_weakrefset.WeakSet object>
     abc negative cache version = 185
     _abc_registry = <_weakrefset.WeakSet object>
     _delete(sql_query, values=None)
     _insert (table_name_or_sql_query, record=None)
     _insert_returning_result(table_name_or_sql_query, record=None)
     _remove (sql_query, values=None)
     _select (query_string, values=None, cursor_factory=None)
     _select_dictresult (sql_query, values=None)
     _select_getresult (sql_query, values=None)
     _select_tuple (sql_query, values)
     _update (query_string, values=None)
     _update_returning_result (query_string, values=None)
     close()
         Close all connections to the database
     connect()
         Create a pool of database connections
     create_tables()
         Create database tables
     delete_data_older_than (days)
         Delete data older that days interval :type days: int :arg days: remove data older than days interval
     duplicate_datablock (taskmanager_id, generation_id, new_generation_id)
         For the given taskmanager id, make a copy of the datablock with given generation id, set the generation id
         for the datablock copy
```

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- new\_generation\_id (int) generation\_id of the new datablock created

#### get connection()

#### get datablock (taskmanager id, generation id)

Return the entire datablock from the dataproduct table for the given taskmanager\_id, generation\_id

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data

### get\_dataproduct (taskmanager\_id, generation\_id, key)

Return the data from the dataproduct table for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- key (string) key for the value

#### get\_header (taskmanager\_id, generation\_id, key)

Return the header from the header table for the given taskmanager id, generation id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- key (string) key for the value

# $\verb|get_last_generation_id| (\textit{taskmanager}_n ame, \textit{taskmanager}_id = None)$

Return last generation id for current task manager or taskmanager w/ task\_manager\_id.

#### **Parameters**

- name (string) task manager name
- taskmanager\_id (string) task manager id

# $\verb"get_metadata" (\textit{taskmanager_id}, \textit{generation\_id}, \textit{key})$

Return the metadata from the metadata table for the given taskmanager id, generation id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- **key** (string) key for the value

#### get\_schema (table=None)

Given the table name return it's schema

### Parameters table (string) - Name of the table

#### get taskmanager (taskmanager name, taskmanager id=None)

Retrieve TaskManager:type taskmanager\_name: string:arg taskmanager\_name: name of taskmanager to retrieve:type taskmanager id: string:arg taskmanager id: id of taskmanager to retrieve

insert (taskmanager\_id, generation\_id, key, value, header, metadata)
Insert data into respective tables for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- **generation\_id** (int) generation\_id of the data
- **key** (string) key for the value
- value (object) Value can be an object or dict
- header (Header) Header for the value
- header Metadata for the value

```
store_taskmanager (name, taskmanager_id)
```

Store TaskManager:type taskmanager\_name: string:arg taskmanager\_name: name of taskmanager to retrieve:type taskmanager\_id: string:arg taskmanager\_id: id of taskmanager to retrieve

```
tables = {'dataproduct': ['taskmanager_id TEXT', 'generation_id INT', 'key TEXT', 'vaupdate (taskmanager_id, generation_id, key, value, header, metadata)
```

Update the data in respective tables for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- key (string) key for the value
- value (object) Value can be an object or dict
- header (Header) Header for the value
- header Metadata for the value

decisionengine.framework.dataspace.datasources.postgresql.generate\_insert\_query(table\_name, keys)

Generate insert query given table name and list of fields

#### **Parameters**

- table\_name (str) Name of the table to insert into
- **keys** List of column names

Keys list

Return type str - insert query

#### **Module contents**

#### **Submodules**

#### decisionengine.framework.dataspace.datablock module

```
class decisionengine.framework.dataspace.datablock.DataBlock (dataspace,
                                                                                           taskman-
                                                                                  ager_id=None,
                                                                                  genera-
                                                                                  tion_id=None, se-
                                                                                  quence id=None)
     Bases: object
     _insert (key, value, header, metadata)
          Insert a new product into database with header and metadata
     _setitem(key, value, header, metadata=None)
          put a product in the database with header and metadata
     _update (key, value, header, metadata)
          Update an existing product in the database with header and metadata
     duplicate()
          Duplicate the datablock and return this new DataBlock. The intent is that at the point the duplication occurs
          there is only information from the sources in the DataBlock. This also increments the generation id of this
          DataBlock.
          TODO: Also update the header and the metadata information TODO: Make this threadsafe
              Return type DataBlock
     get (key, default=None)
          Return the value associated with the key in the database
              Return type dict
     get header (kev)
          Return the Header associated with the key in the database
              Return type Header
     get metadata(key)
          Return the metadata associated with the key in the database
              Return type Metadata
     get_taskmanager_id=None)
          Retrieve TaskManager:type taskmanager_name: string:arg taskmanager_name: name of taskmanager
          to retrieve :type taskmanager_id: string :arg taskmanager_id: id of taskmanager to retrieve :rtype: :obj:
          dict
          The dictionary returned looks like: {'datestamp': datetime.datetime(2017, 12, 20, 17, 37, 17, 503210,
                  tzinfo=psycopg2.tz.FixedOffsetTimezone(offset=-360, name=None)),
               'sequence_id': 135L, 'name': 'AWS_Calculations', 'taskmanager_id': '77B16EB5-C79E-
              45B0-B1B1-37E846692E1D'}
     is_expired(key=None)
          Check if the dataproduct for a given key or any key is expired
```

```
keys()
    mark_expired (expiration_time)
         Set the expiration time for the current generation of the dataproduct and mark it as expired if expira-
         tion time <= current time
    put (key, value, header, metadata=None)
         Put data into the DataBlock
    store_taskmanager(taskmanager_name, taskmanager_id)
         Persist TaskManager, returns sequence number :type taskmanager_name: string :type taskmanager_id:
         :obj: string :rtype: int
exception decisionengine.framework.dataspace.datablock.ExpiredDataError
    Bases: Exception
    Errors due to invalid Metadata
class decisionengine.framework.dataspace.datablock.Header(taskmanager_id,
                                                                    ate time=None, expira-
                                                                    tion time=None, sched-
                                                                    uled create time=None,
                                                                    creator='module',
                                                                    schema_id=None)
    Bases: collections.UserDict
    _abc_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache_version = 185
    _abc_registry = <_weakrefset.WeakSet object>
    default_data_lifetime = 1800
    is valid()
         Check if the Header has minimum required information
    required_keys = {'create_time', 'creator', 'expiration_time', 'scheduled_create_time',
exception decisionengine.framework.dataspace.datablock.InvalidHeaderError
    Bases: Exception
    Errors due to invalid Metadata
exception decisionengine.framework.dataspace.datablock.InvalidMetadataError
    Bases: Exception
    Errors due to invalid Metadata
exception decisionengine.framework.dataspace.datablock.KeyNotFoundError
    Bases: Exception
    Errors due to invalid Metadata
class decisionengine.framework.dataspace.datablock.Metadata(taskmanager_id,
                                                                       state='NEW', gener-
                                                                       ation id=None, gen-
                                                                       eration_time=None,
                                                                      missed update count=0)
    Bases: collections. UserDict.
```

```
_abc_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache_version = 185
    _abc_registry = <_weakrefset.WeakSet object>
    required_keys = {'generation_id', 'generation_time', 'missed_update_count', 'state', '
     set state(state)
         Set the state for the Metadata
    valid_states = {'END_CYCLE', 'METADATA_UPDATE', 'NEW', 'START_BACKUP'}
decisionengine.framework.dataspace.datablock.compress (obj)
    Compress python object :param obj: python object :return: compressed object
decisionengine.framework.dataspace.datablock.decompress(zbytes)
    Decompress zipped byte stream, convert to string. :param zbytes: byte stream :return: uncompressed string
decisionengine.framework.dataspace.datablock.zdumps(obj)
    Pickle and compress :param obj: a python object :return: compressed string
decisionengine.framework.dataspace.datablock.zloads(zbytes)
    Decompress and unpickle If input is not compressed attempts to just unpickle it
         Parameters zbytes – compressed bytes
         Returns returns python object
decisionengine.framework.dataspace.datasource module
class decisionengine.framework.dataspace.datasource.DataSource(config)
    Bases: object
    _abc_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache = <_weakrefset.WeakSet object>
    _abc_negative_cache_version = 185
    _abc_registry = <_weakrefset.WeakSet object>
    abstract close()
         Close all connections to the database
    abstract connect()
         Create a pool of database connections
    abstract create_tables()
         Create database tables
    dataproduct_table = 'dataproduct'
         Name of the dataproduct table
    abstract delete_data_older_than(days)
         Delete data older that interval :type days: long :arg days: remove data older than interval
    abstract duplicate_datablock (taskmanager_id, generation_id, new_generation_id)
         For the given taskmanager id, make a copy of the datablock with given generation id, set the generation id
```

for the datablock copy

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- new\_generation\_id (int) generation\_id of the new datablock created

#### abstract get datablock(taskmanager id, generation id)

Return the entire datablock from the dataproduct table for the given taskmanager id, generation id

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- **generation\_id** (int) generation\_id of the data

#### abstract get\_dataproduct (taskmanager\_id, generation\_id, key)

Return the data from the dataproduct table for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- key (string) key for the value

### abstract get\_header(taskmanager\_id, generation\_id, key)

Return the header from the header table for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- **key** (string) key for the value

### abstract get\_last\_generation\_id(name, taskmanager\_id=None)

Return last generation id for current task manager or taskmanager w/ task\_manager\_id.

#### **Parameters**

- name (string) task manager name
- taskmanager\_id (string) task manager id

# abstract get\_metadata(taskmanager\_id, generation\_id, key)

Return the metadata from the metadata table for the given taskmanager id, generation id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- **key** (string) key for the value

#### abstract get\_schema(table=None)

Given the table name return it's schema

#### **Parameters table** (string) – Name of the table

### abstract get\_taskmanager(taskmanager\_name, taskmanager\_id)

Retrieve TaskManager :type taskmanager\_name: string :arg taskmanager\_name: name of taskmanager to retrieve :type taskmanager\_id: string :arg taskmanager\_id: id of taskmanager to retrieve

#### header table = 'header'

Name of the header table

abstract insert (taskmanager\_id, generation\_id, key, value, header, metadata)

Insert data into respective tables for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- **key** (string) key for the value
- value (object) Value can be an object or dict
- header (Header) Header for the value
- header Metadata for the value

#### metadata\_table = 'metadata'

Name of the metadata table

#### abstract store\_taskmanager(taskmanager\_name, taskmanager\_id)

Store TaskManager:type taskmanager\_name: string:arg taskmanager\_name: name of taskmanager to retrieve:type taskmanager\_id: string:arg taskmanager\_id: id of taskmanager to retrieve

#### taskmanager\_table = 'taskmanager'

Name of the taskmanager table

abstract update (taskmanager\_id, generation\_id, key, value, header, metadata)

Update the data in respective tables for the given taskmanager\_id, generation\_id, key

#### **Parameters**

- taskmanager\_id (string) taskmanager\_id for generation to be retrieved
- generation\_id (int) generation\_id of the data
- **key** (string) key for the value
- value (object) Value can be an object or dict
- header (Header) Header for the value
- header Metadata for the value

#### decisionengine.framework.dataspace.dataspace module

```
class decisionengine.framework.dataspace.dataspace.DataSourceLoader
    Bases: object
    __ds = None
    static create_datasource(module_name, class_name, config)

class decisionengine.framework.dataspace.dataspace.DataSpace(config)
    Bases: object

DataSpace class is collection of datablocks and provides interface to the database used to store the actual data
    __tables_created = False
    Description of tables and their columns
    close()
```

```
delete (taskmanager_id, all_generations=False)
     duplicate_datablock (taskmanager_id, generation_id, new_generation_id)
     get_dataproduct (taskmanager_id, generation_id, key)
     get_header (taskmanager_id, generation_id, key)
     get_last_generation_id (taskmanager_name, taskmanager_id=None)
     get_metadata (taskmanager_id, generation_id, key)
     get_taskmanager (taskmanager_name, taskmanager_id=None)
     insert (taskmanager_id, generation_id, key, value, header, metadata)
     mark_demented (taskmanager_id, keys, generation_id=None)
     mark_expired (taskmanager_id, generation_id, key, expiry_time)
     store_taskmanager (name, id)
     update (taskmanager_id, generation_id, key, value, header, metadata)
exception decisionengine.framework.dataspace.dataspace.DataSpaceConfigurationError
     Bases: Exception
     Errors related to database access
exception decisionengine.framework.dataspace.dataspace.DataSpaceConnectionError
     Bases: Exception
     Errors related to database access
exception decisionengine.framework.dataspace.dataspace.DataSpaceError
     Bases: Exception
     Errors related to database access
exception decisionengine.framework.dataspace.dataspace.DataSpaceExistsError
     Bases: Exception
     Errors related to database access
class decisionengine.framework.dataspace.dataspace.Reaper(config)
     Bases: object
     Reaper provides functionality of periodic deletion of data older than retention_interval in days
     _reaper_loop (delay)
     set state(value)
     get_retention_interval()
     get_state()
     reap()
     set_retention_interval(interval)
     start (delay=0)
         Start thread with an optional delay to start the thread in X seconds
     stop()
```

```
class decisionengine.framework.dataspace.dataspace.Singleton
    Bases: type
    Singleton pattern using Metaclass http://stackoverflow.com/questions/6760685/creating-a-singleton-in-python
    _instances = {}

class decisionengine.framework.dataspace.dataspace.State
    Bases: enum.Enum
    An enumeration.
    ERROR = 7
    IDLE = 1
    RUNNING = 3
    SLEEPING = 4
    STARTING = 2
    STOPPING = 5
```

#### **Module contents**

#### decisionengine.framework.engine package

#### **Submodules**

#### decisionengine.framework.engine.DecisionEngine module

Main loop for Decision Engine. The following environment variable points to decision engine configuration file: DECISION\_ENGINE\_CONFIG\_FILE if this environment variable is not defined the DE-Config.py file from the ``../tests/etc/` directory will be used.

#### \_dispatch (method, params)

Dispatches the XML-RPC method.

XML-RPC calls are forwarded to a registered function that matches the called XML-RPC method name. If no such function exists then the call is forwarded to the registered instance, if available.

If the registered instance has a \_dispatch method then that method will be called with the name of the XML-RPC method and its parameters as a tuple e.g. instance.\_dispatch('add',(2,3))

If the registered instance does not have a \_dispatch method then the instance will be searched to find a matching method and, if found, will be called.

Methods beginning with an '\_' are considered private and will not be called.

```
get_logger()
handle_sighup(signum, frame)
```

```
reaper_start (delay)
    reaper_status()
    reaper_stop()
    reload_config()
    rpc_print_product (product, columns=None, query=None)
    rpc_print_products()
    rpc_reaper_start (delay=0)
         Start the reaper process after 'delay' seconds. Default 0 seconds delay. :type delay: int
    rpc_reaper_status()
    rpc_reaper_stop()
    rpc_reload_config()
    rpc_show_config(channel=None)
    rpc_start_channel (channel)
    rpc_start_channels()
    rpc_status()
    rpc_stop()
    rpc_stop_channel(channel)
    rpc_stop_channels()
    start_channel (channel)
    start_channels()
    stop_channel (channel)
    stop_channels()
class decisionengine.framework.engine.DecisionEngine.RequestHandler(request,
                                                                              client address,
                                                                              server)
    Bases: xmlrpc.server.SimpleXMLRPCRequestHandler
    rpc_paths = ('/RPC2',)
class decisionengine.framework.engine.DecisionEngine.RpcServer(server_address,
                                                                        Re-
                                                                        questHandler-
                                                                        Class)
    Bases: socketserver. Threading MixIn, xmlrpc.server. Simple XMLRPCServer
class decisionengine.framework.engine.DecisionEngine.Worker(task_manager, con-
                                                                    fig)
    Bases: multiprocessing.context.Process
    run()
        Method to be run in sub-process; can be overridden in sub-class
```

### decisionengine.framework.engine.de client module

#### **Module contents**

#### decisionengine.framework.modules package

#### **Submodules**

#### decisionengine.framework.modules.LogicEngine module

```
class decisionengine.framework.modules.LogicEngine.LogicEngine (set_of_parameters)
    Bases: decisionengine.framework.modules.Module.Module
    evaluate (data_block)
```

### decisionengine.framework.modules.Module module

```
class decisionengine.framework.modules.Module.Module(set_of_parameters)
    Bases: object
    get_data_bock()
    get_paramaters()
    set_data_bock(data_block)
```

#### decisionengine.framework.modules.Publisher module

```
class decisionengine.framework.modules.Publisher.Publisher(set_of_parameters)
    Bases: decisionengine.framework.modules.Module.Module
    consumes(name_list)
    publish(data_block=None)
```

#### decisionengine.framework.modules.Source module

```
class decisionengine.framework.modules.Source.Source(set_of_parameters)
    Bases: decisionengine.framework.modules.Module.Module
    acquire()
    produces(name_schema_id_list)
```

### decisionengine.framework.modules.SourceProxy module

```
Fill in data from another channel data block
class decisionengine.framework.modules.SourceProxy.SourceProxy(*args,
                                                                               **kwargs)
     Bases: decisionengine.framework.modules.Source.Source
     Source Proxy Channel configuration using source proxy must have in parameters 'channel_name', defining
     foreign channel name and 'Dataproducts', defining foreign (and optionally local) data keys. See consumes()
     doc. Example of source proxy configuration:
         "AWSJobLimits": { "module": "modules.source_proxy", "name": "SourceProxy", "parameters":
         {"channel_name": "channel_aws_config_data",
                 "Dataproducts":[("aws_instance_limits",
                                                      "Job_Limits")],
                                                                       "retries":
                                                                                    3.
                 "retry_timeout": 20,
             },
         "schedule": 360,
     },
     _get_data(data_block, key)
     acquire()
         Overrides Source class method
     consumes()
         Assumes that self.datakeys has the following structure: is a list of tuples or singletons:
             (data product name, data product name translation), .... ] or [ data product name, ....
     must_have = ('channel_name', 'Dataproducts')
     produces()
         Assumes that self.datakeys has the following structure or
decisionengine.framework.modules.SourceProxy.main()
     Call this a a test unit or use as CLI of this module
decisionengine.framework.modules.SourceProxy.module_config_info()
     print this module configuration information
decisionengine.framework.modules.SourceProxy.module config template()
     print a template for this module configuration data
decisionengine.framework.modules.Transform module
class decisionengine.framework.modules.Transform.Transform(set_of_parameters)
     Bases: decisionengine.framework.modules.Module.Module
     consumes (name_list)
     produces (name_schema_id_list)
```

transform()

### decisionengine.framework.modules.de logger module

```
Looger to use in all modules
```

#### **Parameters**

- log\_file\_name (str) log file name
- max\_file\_size (int) maximal size of log file. If reached save and start new log.
- max\_backup\_count (int) start rotaion after this number is reached

Return type logging. Logger - rotating file logger

decisionengine.framework.modules.de\_logger.set\_stream\_logging(logger\_name=") This is for debugging. Set stream logging for logger.

Parameters logger\_name (str) - logger name

Return type logging.Logger

#### **Module contents**

#### decisionengine.framework.taskmanager package

#### **Submodules**

# decisionengine.framework.taskmanager.TaskManager module

```
Task Manager
```

```
\textbf{class} \ \texttt{decisionengine.framework.taskmanager.TaskManager.Channel} \ (\textit{channel\_dict}) \\ \textbf{Bases: object}
```

Decision Channel. Instantiates workers according to channel configuration

```
class decisionengine.framework.taskmanager.TaskManager.TaskManager(name,
```

task\_manager\_id, generation\_id, channel\_dict, global config)

Bases: object

Task Manager

data\_block\_put (data, header, data\_block)

Put data into data block

#### **Parameters**

• data (dict) - key, value pairs

```
• header (Header) - data header

    data_block (DataBlock) – data block

decision_cycle()
    Decision cycle to be run periodically (by trigger)
do backup()
    Duplicate current data block and return its copy
        Return type DataBlock
get_state()
offline_task_manager(current_data_block)
    offline and stop task manager
run()
    Task Manager main loop
run_logic_engine (data_block=None)
    Run Logic Engine.
        Parameters data_block (DataBlock) - data block
run_publishers (actions, facts, data_block=None)
    Run Publishers in main process.
        Parameters data block (DataBlock) - data block
run source (src)
    Get the data from source and put it into the data block
        Parameters src(Worker) - source Worker
run_transform(transform, data_block)
    Run a transform
        Parameters
            • transform (Worker) - source Worker
            • data_block (DataBlock) - data block
run_transforms (data_block=None)
    Run transforms. So far in main process.
        Parameters data_block (DataBlock) - data block
set_state(state)
start_sources (data_block=None)
    Start sources, each in a separate thread
        Parameters data_block (DataBlock) - data block
stop_task_manager()
    signal task manager to stop
wait_for_all(events_done)
    Wait for all sources or transforms to finish
        Parameters events_done (list) - list of events to wait for
wait_for_any (events_done)
    Wait for any sources to finish
```

#### **Parameters** events\_done (list) – list of events to wait for

Provides interface to loadable modules an events to sycronise execution

DEFAULT SCHEDULE = 300

decisionengine.framework.taskmanager.TaskManager.log\_exception(logger, header\_message)

#### **Module contents**

### decisionengine.framework.util package

#### **Submodules**

# decisionengine.framework.util.tsort module

See:

https://en.wikipedia.org/wiki/Topological\_sorting

Kahn's topological sorting algorithm

L Empty list that will contain the sorted elements S Set of all nodes with no incoming edge while S is non-empty do remove a node n from S add n to tail of L for each node m with an edge e from n to m do

remove edge e from the graph if m has no other incoming edges then

insert m into S

**if graph has edges then** return error (graph has at least one cycle)

else return L (a topologically sorted order)

```
decisionengine.framework.util.tsort.tsort(graph)
```

Function implementing Kahn's topological sorting algorithm returns two lists: sorted list and cyclic lost (if graph is acyclic second list is always None)

Return type list

#### Module contents

#### Module contents

#### Module contents

# 3.2 Indices and tables

- genindex
- modindex
- · search

3.2. Indices and tables 27

# **CHAPTER**

# **FOUR**

# **INDICES AND TABLES**

- genindex
- modindex
- search

# **PYTHON MODULE INDEX**

```
d
                                         decisionengine.framework.util, 27
                                         decisionengine.framework.util.tsort, 27
decisionengine, 27
decisionengine.framework, 27
decisionengine.framework.configmanager,
decisionengine.framework.configmanager.ConfigManager,
decisionengine.framework.dataspace, 21
decisionengine.framework.dataspace.datablock,
decisionengine.framework.dataspace.datasource,
decisionengine.framework.dataspace.datasources,
decisionengine.framework.dataspace.datasources.postgresql,
decisionengine.framework.dataspace.dataspace,
decisionengine.framework.engine, 23
decisionengine.framework.engine.de client,
decisionengine.framework.engine.DecisionEngine,
decisionengine.framework.modules, 25
decisionengine.framework.modules.de logger,
decisionengine.framework.modules.LogicEngine,
decisionengine.framework.modules.Module,
decisionengine.framework.modules.Publisher,
decisionengine.framework.modules.Source,
decisionengine.framework.modules.SourceProxy,
decisionengine.framework.modules.Transform,
decisionengine.framework.taskmanager,
decisionengine.framework.taskmanager.TaskManager,
```

32 Python Module Index

# **INDEX**

Symbols			attribute), 16	
_Postgresqlquery()	(decisio-	_abc_r	registry	(decisio-
nengine.framework.dataspace.datasot method), 12	urces.postgi		anribute), 17	
_abc_cache	(decisio-	_abc_r	registry	(decisio-
nengine.framework.dataspace.datable attribute), 16			attribute), 17	space.datasource.DataSource
_abc_cache	(decisio-	_abc_r	registry	(decisio-
nengine.framework.dataspace.datable attribute), 16			attribute), 12	space.datasources.postgresql.Postgresql
_abc_cache	(decisio-	_delet	e () (decisionengine.fram)	nework.dataspace.datasources.postgresq
nengine.framework.dataspace.datasot		ource	method), 12	
attribute), 17		_dispa		(decisio-
_abc_cache	(decisio-			ne.DecisionEngine.DecisionEngine
nengine.framework.dataspace.datasot	urces.postgi	resql.Post	g method), 21	
attribute), 12		_ds (de	cisionengine.jramework.a	ataspace.dataspace.DataSourceLoader
_abc_negative_cache	(decisio-		attribute), 19	
nengine.framework.dataspace.datable	ock.Header	_get_d	lata()	(decisio-
attribute), 16			nengine.jramework.moa	ules.SourceProxy.SourceProxy
_abc_negative_cache	(decisio-		method), 24	
nengine.framework.dataspace.datablo	ock.Metadat	a-inser	rt () (decisionengine.fran	nework.dataspace.datablock.DataBlock
attribute), 17			metnoa), 15	
_abc_negative_cache	(accisio			nework.dataspace.datasources.postgresq
nengine.framework.dataspace.datasoi	urce.DataSc	ource	<pre>method), 12 rt_returning_resul</pre>	t() (decisio-
attribute), 17		_IIISeI		space.datasources.postgresql.Postgresql
_abc_negative_cache	(decisio-			space.aatasources.posigresqt.1 osigresqt
nengine.framework.dataspace.datasot attribute), 12	urces.postgi	resql.Post	grestion), 12	(decisio-
			nengine framework data	espace.dataspace.Singleton
_abc_negative_cache_version	(decisio-		attribute), 21	space.uaraspace.singleton
nengine.framework.dataspace.datable	оск.Неаает	reape	er_loop()	(decisio-
attribute), 16	(decisio-		nengine.framework.data	(
_abc_negative_cache_version	*	ta	method), 20	
attribute), 17	ck.weiaaai	remov	/ /	nework.dataspace.datasources.postgresql
aha nogativo gagho vorgion	(decisio-		method), 12	
nengine framework dataspace datasp	urce DataSo	, , , , , , , , , , ,	t () (decisionengine.fran	nework.dataspace.datasources.postgresql
attribute), 17			metnoa), 12	
aha nagatira asaha manaian	(decisio-	_selec	t_dictresult()	(decisio-
nengine.framework.dataspace.datasot attribute), 12	urces.postgi	resql.Post	g pengine.framework.data method), 12	space.datasources.postgresql.Postgresql
_abc_registry	(decisio-	_selec	t_getresult()	(decisio-
nengine.framework.dataspace.datable	`		nengine.framework.data	space.datasources.postgresql.Postgresql

```
(decisio-
                                  method), 12
                                                                                                                                                                                                                    consumes()
 _select_tuple()
                                                                                                                                                                            (decisio-
                                                                                                                                                                                                                                                       nengine.framework.modules.SourceProxy.SourceProxy
                                  nengine.framework.dataspace.datasources.postgresql.Postgrandhod), 24
                                  method), 12
                                                                                                                                                                                                                    consumes()
                                                                                                                                                                                                                                                                                                                                                                                                (decisio-
_set_state()
                                                                                                                                                                           (decisio-
                                                                                                                                                                                                                                                       nengine.framework.modules.Transform.Transform
                                  nengine.framework.dataspace.dataspace.Reaper
                                                                                                                                                                                                                                                       method), 24
                                  method), 20
                                                                                                                                                                                                                    create() (decisionengine.framework.configmanager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigMan
                                                                                                                                                                                                                                                       static method), 11
setitem()
                                                                                                                                                                            (decisio-
                                  nengine.framework.dataspace.datablock.DataBlockreate datasource()
                                                                                                                                                                                                                                                                                                                                                                                                 (decisio-
                                  method), 15
                                                                                                                                                                                                                                                       nengine. framework. dataspace. dataspace. Data Source Loader
_tables_created
                                                                                                                                                                            (decisio-
                                                                                                                                                                                                                                                       static method), 19
                                   nengine.framework.dataspace.dataspace.DataSpacereate_tables()
                                                                                                                                                                                                                                                                                                                                                                                                (decisio-
                                                                                                                                                                                                                                                       nengine.framework.dataspace.datasource.DataSource
                                  attribute), 19
_update() (decisionengine.framework.dataspace.datablock.DataBlonethod), 17
                                   method), 15
                                                                                                                                                                                                                    create_tables()
                                                                                                                                                                                                                                                                                                                                                                                                 (decisio-
_update() (decisionengine.framework.dataspace.datasources.postgrengirRoßigmenybrk.dataspace.datasources.postgresql.Postgresql
                                   method), 12
                                                                                                                                                                                                                                                       method), 12
_update_returning_result()
                                                                                                                                                                           (decisio-
                                  nengine.framework.dataspace.datasources.postgresql.Postgresql
                                                                                                                                                                                                                    data_block_put()
                                                                                                                                                                                                                                                                                                                                                                                                (decisio-
                                                                                                                                                                                                                                                       nengine.framework.taskmanager.TaskManager.TaskManager
Α
                                                                                                                                                                                                                                                       method), 25
acquire() (decisionengine.framework.modules.Source.Source.Source
                                                                                                                                                                                                                                                                                                     (class
                                                                                                                                                                                                                                                                                                                                                           in
                                                                                                                                                                                                                                                                                                                                                                                                   decisio-
                                   method), 23
                                                                                                                                                                                                                                                       nengine.framework.dataspace.datablock),
acquire()(decisionengine.framework.modules.SourceProxy.SourcdBroxy
                                   method), 24
                                                                                                                                                                                                                    dataproduct_table
                                                                                                                                                                                                                                                                                                                                                                                                 (decisio-
                                                                                                                                                                                                                                                       nengine.framework.dataspace.datasource.DataSource
С
                                                                                                                                                                                                                                                       attribute), 17
                                                                                                                                                                                                                   DataSource
                                                                                                                                                                                                                                                                                                         (class
                                                                                                                                                                                                                                                                                                                                                             in
                                                                                                                                                                                                                                                                                                                                                                                                   decisio-
Channel
                                                                          (class
                                                                                                                                                                               decisio-
                                                                                                                                                                                                                                                       nengine.framework.dataspace.datasource),
                                  nengine.framework.taskmanager.TaskManager),
                                                                                                                                                                                                                                                        17
                                   25
                                                                                                                                                                                                                   DataSourceLoader
                                                                                                                                                                                                                                                                                                                              (class
                                                                                                                                                                                                                                                                                                                                                                                                   decisio-
                                                                                                                                                                           (decisio-
check_keys()
                                  nengine. framework. config {\it Manager. Config Manager. Config Manager. Config {\it Manager. Config Manager. Config Manager. Config {\it Manager. Config {\it Manager. Config Manager. Config {\it Manager.
                                                                                                                                                                                                                                                                                                     (class
                                                                                                                                                                                                                                                                                                                                                           in
                                                                                                                                                                                                                                                                                                                                                                                                   decisio-
close () (decisionengine.framework.dataspace.datasource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassource.abtassour
                                                                                                                                                                                                                                                       nengine.framework.dataspace.dataspace),
                                   method), 17
close() (decisionengine.framework.dataspace.datasources.postgresql.Postgresql
                                                                                                                                                                                                                    DataSpaceConfigurationError, 20
close() (decisionengine.framework.dataspace.dataspace.DataspaceeConnectionError, 20
                                                                                                                                                                                                                   DataSpaceError, 20
                                  method), 19
                                                                                                                                                                                                                   DataSpaceExistsError, 20
compress()
                                                                                 (in
                                                                                                                       module
                                                                                                                                                                               decisio-
                                                                                                                                                                                                                    decision_cycle()
                                                                                                                                                                                                                                                                                                                                                                                                (decisio-
                                  nengine.framework.dataspace.datablock),
                                                                                                                                                                                                                                                       nengine.framework.taskmanager.TaskManager.TaskManager
                                                                                                                                                                                                                                                       method), 26
                                                                                                                                                                               decisio-
ConfigManager
                                                                                               (class
                                                                                                                                              in
                                  \textit{nengine.} \textit{framework.} \textit{configManager.} \textit{ConfigManager.} \textbf{ConfigManager.} \textbf{
                                                                                                                                                                                                                                     module, 27
connect () (decisionengine.framework.dataspace.datasouPree:DataSouFreeine
                                                                                                                                                                                                                                                                                                                       (class
                                                                                                                                                                                                                                                                                                                                                                                                   decisio-
                                                                                                                                                                                                                                                       nengine.framework.engine.DecisionEngine),
                                  method), 17
connect () (decisionengine.framework.dataspace.datasources.postgresql.Postgresql
                                                                                                                                                                                                                    decisionengine.framework
                                  method), 12
                                                                                                                                                                                                                                     module, 27
                                                                                                                                                                            (decisio-
consumes()
                                  \textit{nengine.} \textit{framework.} \textit{modules.} \textit{Publisher.} \textit{Publisher} \text{ decisionengine.} \textit{framework.} \textit{configmanager}
                                                                                                                                                                                                                                     module, 12
                                  method), 23
                                                                                                                                                                                                                    decisionengine.framework.configmanager.ConfigManage
```

```
module, 11
                                                                                                                                                                   delete_data_older_than()
                                                                                                                                                                                                                                                                                                       (decisio-
decisionengine.framework.dataspace
                                                                                                                                                                                              nengine.framework.dataspace.datasource.DataSource
decisionengine.framework.dataspace.databdeckte_data_older_than()
                                                                                                                                                                                                                                                                                                       (decisio-
              module, 15
                                                                                                                                                                                              nengine.framework.dataspace.datasources.postgresql.Postgresql
decisionengine.framework.dataspace.datasource method), 12
                                                                                                                                                                                                                                                                                                       (decisio-
             module, 17
                                                                                                                                                                   do backup()
decisionengine.framework.dataspace.datasources nengine.framework.taskmanager.TaskManager
              module, 14
                                                                                                                                                                                              method), 26
decisionengine.framework.dataspace.datasduptesapestgresql
                                                                                                                                                                                                                                                                                                       (decisio-
             module, 12
                                                                                                                                                                                              nengine.framework.dataspace.datablock.DataBlock
decisionengine.framework.dataspace.dataspace
                                                                                                                                                                                              method), 15
             module, 19
                                                                                                                                                                   duplicate_datablock()
                                                                                                                                                                                                                                                                                                       (decisio-
                                                                                                                                                                                              nengine.framework.dataspace.datasource.DataSource
 decisionengine.framework.engine
             module, 23
                                                                                                                                                                                               method), 17
decisionengine.framework.engine.de_clienduplicate_datablock()
                                                                                                                                                                                                                                                                                                       (decisio-
                                                                                                                                                                                              nengine.framework.dataspace.datasources.postgresql.Postgresql
decisionengine.framework.engine.DecisionEngine method), 12
                                                                                                                                                                   duplicate_datablock()
             module, 21
                                                                                                                                                                                                                                                                                                       (decisio-
 decisionengine.framework.modules
                                                                                                                                                                                              nengine.framework.dataspace.dataspace.DataSpace
              module, 25
                                                                                                                                                                                              method), 20
decisionengine.framework.modules.de_logger
              module, 25
\texttt{decisionengine.} framework. \texttt{modules.LogicEng} \\ \texttt{ROR} (\textit{decisionengine.} \textit{framework.} \textit{dataspace.} \textit{dat
              module, 23
                                                                                                                                                                                               attribute), 21
decisionengine.framework.modules.Module evaluate()
                                                                                                                                                                                                                                                                                                       (decisio-
             module, 23
                                                                                                                                                                                              nengine.framework.modules.LogicEngine.LogicEngine
decisionengine.framework.modules.Publisher
                                                                                                                                                                                              method), 23
             module, 23
                                                                                                                                                                   ExpiredDataError, 16
decisionengine.framework.modules.Source
                                                                                                                                                                    G
              module, 23
{\tt decisionengine.framework.modules.SourceP} {\tt generate\_insert\_query()} \ \textit{(in module decisionengine.framework)} \\
                                                                                                                                                                                              nengine.framework.dataspace.datasources.postgresql),
decisionengine.framework.modules.Transform
              module, 24
                                                                                                                                                                   get () (decisionengine.framework.dataspace.datablock.DataBlock
decisionengine.framework.taskmanager
                                                                                                                                                                                              method), 15
module, 27 get_channels() (decisio-decisionengine.framework.taskmanager.TaskManager.framework.configmanager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.ConfigManager.Config
              module, 25
                                                                                                                                                                                              method), 11
decisionengine.framework.util
                                                                                                                                                                   get connection()
                                                                                                                                                                                                                                                                                                       (decisio-
             module, 27
                                                                                                                                                                                              nengine.framework.dataspace.datasources.postgresql.Postgresql
decisionengine.framework.util.tsort
                                                                                                                                                                                              method), 13
              module, 27
                                                                                                                                                                   get data bock()
                                                                                                                                                                                                                                                                                                       (decisio-
                                                                                               module
                                                                                                                                       decisio-
decompress()
                                                                     (in
                                                                                                                                                                                              nengine.framework.modules.Module.Module
                           nengine.framework.dataspace.datablock),
                                                                                                                                                                                              method), 23
                                                                                                                                                                                                                                                                                                       (decisio-
                                                                                                                                                                   get_datablock()
                                                                                                                                    (decisio-
default_data_lifetime
                                                                                                                                                                                               nengine.framework.dataspace.datasource.DataSource
                           nengine.framework.dataspace.datablock.Header
                                                                                                                                                                                              method), 18
                           attribute), 16
                                                                                                                                                                   get_datablock()
                                                                                                                                                                                                                                                                                                       (decisio-
                                                                                                                                    (decisio-
DEFAULT_SCHEDULE
                                                                                                                                                                                              nengine.framework.dataspace.datasources.postgresql.Postgresql
                           nengine.framework.taskmanager.TaskManager.Worker
                                                                                                                                                                                               method), 13
attribute), 27 \\ \texttt{get\_dataproduct()} \\ \texttt{decision} \\ \texttt{delete()} \\ \textit{(decisionengine.framework.dataspace.dataspace.dataspace.DataSpace.pataSpace.gataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspace.dataspac
                           method), 19
```

method), 18

```
get_dataproduct()
                                            (decisio- get_retention_interval()
                                                                                                   (decisio-
        nengine.framework.dataspace.datasources.postgresql.Postgresqgine.framework.dataspace.dataspace.Reaper
        method), 13
                                                               method), 20
get_dataproduct()
                                            (decisio- get schema()
                                                                                                   (decisio-
        nengine.framework.dataspace.dataspace.DataSpace
                                                               nengine.framework.dataspace.datasource.DataSource
        method), 20
                                                               method), 18
                                            (decisio- get schema()
                                                                                                   (decisio-
get_global_config()
        nengine.framework.configmanager.ConfigManager.ConfigManagine.framework.dataspace.datasources.postgresql.Postgresql
        method), 11
                                                               method), 13
                                            (decisio- get_state()
get_header()
                                                                                                   (decisio-
                                                               nengine.framework.dataspace.dataspace.Reaper
        nengine.framework.dataspace.datablock.DataBlock
        method), 15
                                                               method), 20
get_header()
                                            (decisio- get_state()
                                                                                                   (decisio-
        nengine.framework.dataspace.datasource.DataSource
                                                               nengine.framework.taskmanager.TaskManager.TaskManager
                                                                method), 26
        method), 18
get_header()
                                            (decisio- get_taskmanager()
                                                                                                   (decisio-
        nengine.framework.dataspace.datasources.postgresql.Postgresqkine.framework.dataspace.datablock.DataBlock
                                                               method), 15
get_header()
                                            (decisio- get_taskmanager()
                                                               nengine.framework.dataspace.datasource.DataSource
        nengine.framework.dataspace.dataspace.DataSpace
        method), 20
                                                               method), 18
get_last_generation_id()
                                            (decisio- get_taskmanager()
                                                                                                   (decisio-
        nengine. framework. data space. data source. Data Source
                                                               nengine.framework.dataspace.datasources.postgresql.Postgresql
        method), 18
                                                               method), 13
                                            (decisio- get_taskmanager()
get_last_generation_id()
                                                                                                   (decisio-
        nengine.framework.dataspace.datasources.postgresql.Postgresqline.framework.dataspace.dataspace.DataSpace
        method), 13
                                                               method), 20
                                            (decisio-
get_last_generation_id()
        nengine.framework.dataspace.dataspace.DataSpace
        method), 20
                                                      handle sighup()
                                                                                                   (decisio-
                                            (decisio-
                                                               nengine.framework.engine.DecisionEngine.DecisionEngine
         nengine.framework.engine.DecisionEngine.DecisionEngine method), 21
        method), 21
                                                      Header
                                                                        (class
                                                                                        in
                                                                                                   decisio-
                               module
get_logger()
                      (in
                                             decisio-
                                                               nengine.framework.dataspace.datablock),
         nengine.framework.modules.de logger),
                                                                16
                                                      header table
get_metadata()
                                            (decisio-
                                                                nengine.framework.dataspace.datasource.DataSource
        nengine.framework.dataspace.datablock.DataBlock
                                                                attribute), 18
        method), 15
                                            (decisio-
get_metadata()
        nengine. framework. dataspace. datas our ce. Data Sour {\tt Selection} (decision engine. framework. dataspace. dataspace. State)
        method), 18
                                                               attribute), 21
get_metadata()
                                            (decisio-
        tadata() (decisio- insert() (decisionengine.framework.dataspace.datasource.DataSource nengine.framework.dataspace.datasources.postgresql.Postgresql, 19
        method), 13
                                                      insert () (decisionengine.framework.dataspace.datasources.postgresql.I
                                            (decisio-
get_metadata()
                                                               method), 13
        nengine. framework. dataspace. dataspace. DataSpace \\ {\tt Insert()(decisionengine. framework. dataspace. dataspace. DataSpace)} \\ \\
        method), 20
                                                               method), 20
get_paramaters()
                                            (decisio-
                                                      InvalidHeaderError, 16
        nengine.framework.modules.Module.Module
                                                      InvalidMetadataError, 16
        method), 23
                                                      is_expired()
                                                                                                   (decisio-
                                            (decisio-
get_produces()
                                                               nengine.framework.dataspace.datablock.DataBlock
        nengine.framework.configmanager.ConfigManager.ConfigManager.html
        method), 11
```

```
is_updated()
                                       (decisio-
                                                    decisionengine.framework.dataspace.datasources,
       nengine.framework.configmanager.ConfigManager.ConfigMahager
                                                    decisionengine.framework.dataspace.datasources.
       method), 11
                                       (decisio-
is_valid()
       nengine.framework.dataspace.datablock.Header
                                                    decisionengine.framework.dataspace.dataspace,
       method), 16
                                                    decisionengine.framework.engine, 23
K
                                                    decisionengine.framework.engine.de client,
KeyNotFoundError, 16
keys()(decisionengine.framework.dataspace.datablock.DataBtdekisionengine.framework.engine.DecisionEngine,
        method), 15
                                                    decisionengine.framework.modules, 25
                                                    decisionengine.framework.modules.de_logger,
load() (decisionengine.framework.configmanager.ConfigManager.ConfigManager
                                                    decisionengine.framework.modules.LogicEngine,
        method), 11
log_exception()
                      (in
                             module
                                       decisio-
                                                    decisionengine.framework.modules.Module,
       nengine.framework.taskmanager.TaskManager),
                                                    decisionengine.framework.modules.Publisher,
LogicEngine
                    (class
                               in
                                       decisio-
       nengine.framework.modules.LogicEngine),
                                                    decisionengine.framework.modules.Source,
M
                                                    decisionengine.framework.modules.SourceProxy,
main()
               (in
                         module
                                       decisio-
                                                    decisionengine.framework.modules.Transform,
       nengine.framework.modules.SourceProxy),
        24
                                                    decisionengine.framework.taskmanager,
mark_demented()
                                       (decisio-
        nengine.framework.dataspace.dataspace.DataSpace
                                                    decisionengine.framework.taskmanager.TaskManage
       method), 20
mark_expired()
                                       (decisio-
                                                    decisionengine.framework.util, 27
       nengine.framework.dataspace.datablock.DataBlock
                                                    decisionengine.framework.util.tsort,
       method), 16
                                                        27
mark_expired()
                                       (decisio-
       nengine. framework. data space. data space. Data Space {\tt odule}
                                                                (class
                                                                                        decisio-
                                                        nengine.framework.modules.Module), 23
       method), 20
                                                module_config_info()
                                                                          (in module
Metadata
                 (class
                                       decisio-
                                                        nengine.framework.modules.SourceProxy),
        nengine.framework.dataspace.datablock),
        16
                                                module_config_template() (in module decisio-
                                       (decisio-
metadata_table
                                                        nengine.framework.modules.SourceProxy), 24
       nengine.framework.dataspace.datasource.DataSource
                                                must have (decisionengine.framework.modules.SourceProxy.SourceProx
       attribute), 19
                                                        attribute), 24
module
    decisionengine, 27
                                                \bigcirc
    decisionengine.framework, 27
    {\tt decisionengine.framework.configmanage} {\tt ffline\_task\_manager()}
                                                        nengine.framework.taskmanager.TaskManager.TaskManager
    decisionengine.framework.configmanager.ConfigManager,
                                                Р
    decisionengine.framework.dataspace,
                                                Postgresql
                                                                   (class
                                                                               in
                                                                                        decisio-
    decisionengine.framework.dataspace.datablocknengine.framework.dataspace.datasources.postgresql),
    decisionengine.framework.dataspace.datasducee()
                                                                                       (decisio-
        17
                                                        nengine.framework.modules.Source.Source
```

	method), 23		method), 22	
produce		(decisio- rpc_rea	aper_status()	(decisio-
	nengine. framework. modules. Source Practical Control of the property of the	oxy.SourceProxy	nengine. framework. engine. Decision E	Ingine.DecisionEngine
	method), 24		method), 22	
produce		(decisio- rpc_rea		(decisio-
	nengine.framework.modules.Transformethod), 24	m.Transform	nengine.framework.engine.DecisionE method), 22	Ingine.DecisionEngine
publis	h () (decisionengine.framework.modul	les.Publishem <u>P</u> ub <u>li</u> ske	droad_config()	(decisio-
Publis	method), 23 her (class in		nengine.framework.engine.DecisionE method), 22	Ingine.DecisionEngine
	nengine.framework.modules.Publisher	r), rpc_sh	ow_config()	(decisio-
	23		nengine. framework. engine. Decision E	Ingine.DecisionEngine
put()(	decisionengine.framework.dataspace.d	atablock.DataBlock	method), 22	
	method), 16	rpc_sta	art_channel()	(decisio-
R			nengine.framework.engine.DecisionE method), 22	Ingine.DecisionEngine
reap()	(decision engine. framework. data space.	dataspace.R&@perst	art_channels()	(decisio-
	method), 20		nengine.framework.engine.DecisionE method), 22	Ingine.DecisionEngine
_	nengine.framework.dataspace.dataspa		atus()	(decisio-
	20	,	nengine. framework. engine. Decision E	Ingine.DecisionEngine
reaper_	_start()	(decisio-	method), 22	
	nengine.framework.engine.DecisionEn	ngine.Decisiชิเ <mark>ชิโรกรูร</mark> ีก็ <del>é</del>	pp ()	(decisio-
	method), 22		nengine.framework.engine.DecisionE	Ingine.DecisionEngine
	_status()	(	method), 22	
	nengine.framework.engine.DecisionEn	ngine.Decisiชิเชียนสูรี่ค็	op_channel()	(decisio-
	method), 22		nengine.framework.engine.DecisionE	ngine.DecisionEngine
reaper_	_stop()	(	method), 22	(decisio
	nengine.framework.engine.DecisionEn	ngine.Decisi <del>bleEngine</del>	pp_cnamers() nengine.framework.engine.DecisionE	(decisio-
	method), 22			ngine.DecisionEngine
	() (decisionengine.framework.configm	<i>anager.ConjigManag</i> RpcSer	en <del>Conµgwa</del> nager ver (class in	decisio-
	method), 11 _config()	(decisio-	nengine.framework.engine.DecisionE	
	_confrg() nengine.framework.engine.DecisionEi	•	9 0	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	method), 22		decisionengine.framework.engine.Deci	isionEngine.Worker
	tHandler (class in	decisio-	method), 22	
	nengine.framework.engine.DecisionEn	ngine), run()(	decisionengine.framework.taskmanage method), 26	r.TaskManager.TaskManage
require	ed_keys		gic_engine()	(decisio-
	nengine.framework.dataspace.datablo attribute), 16	(	nengine.framework.taskmanager.Task method), 26	Manager.TaskManager
	ed_keys	(decisio- run_pul	blishers()	(decisio-
_	nengine.framework.dataspace.datablo attribute), 17		nengine.framework.taskmanager.Task method), 26	Manager.TaskManager
	ths (decisionengine.framework.engine	e.DecisionEnvine_Red	WEST <b>H</b> andler	(decisio-
1	attribute), 22		nengine.framework.taskmanager.Task	Manager.TaskManager
rpc_pr	int_product()	(decisio-	method), 26	
	nengine.framework.engine.DecisionEn		ansform()	(decisio-
	method), 22		nengine.framework.taskmanager.Task	Manager.TaskManager
rpc_pr	int_products()	(decisio-	method), 26	
	nengine. framework. engine. Decision Engine. The contract of	ngine.Decisiอิฟ <b>E</b> ngiñe	ansforms()	(decisio-
	method), 22		nengine.framework.taskmanager.Task	Manager.TaskManager
rpc_rea	aper_start() nengine.framework.engine.DecisionEi	(decisio-	method), 26 C (decisionengine framework dataspac	ce datasnace State
	nengine.jramework.engine.DecisionEi	ngine.Decisiอพยหรูเพ	e (accisionenzine.jramework.aaaspac	с.аашэрисс.ыше

attribute), 21	<pre>stop_task_manager()</pre>
S	method), 26
set_data_bock() (decisio-	STOPPED (decisionengine.framework.dataspace.dataspace.State
nengine.framework.modules.Module.Module	attribute), 21
method), 23	STOPPING (decisionengine.framework.dataspace.dataspace.State attribute), 21
set_logging() (in module decisio-	store_taskmanager() (decisio-
nengine.framework.modules.de_logger), 25	nengine.framework.dataspace.datablock.DataBlock
set_retention_interval() (decisio-	method), 16
nengine.framework.dataspace.dataspace.Reaper	
method), 20	nengine.framework.dataspace.datasource.DataSource
set_state() (decisio-	method), 19
nengine.framework.dataspace.datablock.Metada	tastore_taskmanager() (decisio-
method), 17	nengine. framework. data space. data sources. postgresql. Postgresquare for the sum of the state of the sta
set_state() (decisio-	method), 14
nengine. framework. task manager. Task Manager. To the property of the prope	uskManageraskmanager() (decisio-
method), 26	nengine.framework.dataspace.dataspace.DataSpace
set_stream_logging() (in module decisio-	method), 20
nengine.framework.modules.de_logger), 25	Т
Singleton (class in decisio-	
nengine.framework.dataspace.dataspace), 20	tables (decisionengine.framework.dataspace.datasources.postgresql.Po attribute), 14
SLEEPING (decisionengine.framework.dataspace.dataspa	
attribute), 21	nengine.framework.taskmanager.TaskManager),
Source (class in decisio-	25
nengine.framework.modules.Source), 23	
SourceProxy (class in decisionengine.framework.modules.SourceProxy),	nengine.framework.dataspace.datasource.DataSource attribute), 19
nengine.jramework.modutes.sourcerroxy), 24	Transform (class in decisio-
start() (decisionengine.framework.dataspace.dataspace	
method), 20	24
	transform() (decisio-
_ ``	sionEngine nengine.framework.modules.Transform.Transform
method), 22	method), 24
start_channels() (decisio-	tsort() (in module decisio-
nengine. framework. engine. Decision E	sionEngine nengine.framework.util.tsort), 27
method), 22	U
start_sources() (decisio-	•
nengine.framework.taskmanager.TaskManager.Ta	$ask {\it Monager}$ ) (decisionengine.framework.dataspace.datasource.DataSource
method), 26	method), 19
STARTING (decisionengine.framework.dataspace.dataspa	CeState = () (decisionengine.framework.dataspace.datasources.postgresql.
attribute), 21 State (class in decisio-	method), 14
State (class in decisio- nengine.framework.dataspace.dataspace),	update() (decisionengine.framework.dataspace.dataspace.DataSpace
21	method), 20
stop () (decisionengine.framework.dataspace.dataspace.	Rymer
method), 20	
stop_channel() (decisio-	valid_states (decisio-
nengine.framework.engine.DecisionEngine.Decis	nengine.framework.dataspace.datablock.Metadata sionEngine_attribute)
method), 22	validate_channel() (decisio-
stop_channels() (decisio-	nengine framework configurances ConfigManager ConfigMana
nengine.framework.engine.DecisionEng	sionEngine method), 11

# W

```
wait_for_all()
                                            (decisio-
         nengine. framework. task manager. Task Manager. Task Manager
        method), 26
                                            (decisio-
wait_for_any()
         nengine. framework. task manager. Task Manager. Task Manager
         method), 26
Worker
                  (class
                                 in
                                             decisio-
         nengine.framework.engine.DecisionEngine),
Worker
                  (class
                                 in
                                             decisio-
         nengine. framework. taskmanager. TaskManager),
Ζ
zdumps()
                   (in
                              module
                                             decisio-
         nengine.framework.dataspace.datablock),
                                             decisio-
zloads()
                   (in
                              module
         nengine.framework.dataspace.datablock),
```