# decisionengine

Release 1.7.1.dev34+ge95071fd

**Fermilab** 

# **CONTENTS**

1	Release Notes	3
2	Developer Documentation	25
3	Jenkins CI pipeline	27
4	Source code	33
5	Indices and tables	99
Ру	Python Module Index	101
Index		105

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 Notes

HEPCloud's Decision Engine release notes.

The latest release is the designated production release. Decision Engine will support also N-1. New feature development will happen in the development branch and go in the next (N+1) release.

# 1.1.1 Release 1.7.0

This release features:

- New produces-consumes structure using decorators. This will improve the code quality, improving static checks and reducing the lines of code by removing repetitive boilerplates, especially in the modules.
- Added structured logging. Improved python logging and adoption of structured logs format that will increase the semantinc content of the messages and ease the export of information for dashboards and Elastic Search.
- Added SQLAlchemy object-relational mapper to increase the testability of DB interactions and to allow different database backends. Switching between datasource backends requires dropping all objects if you wish to reuse the tablespace.
- Packaging via setuptools for both decisionengine and decisionengine\_modules: Dependencies are not yet fully listed in the RPMs.
- A new, optional, configuration parameter called "channel\_name" is available. "channel\_name" is one of the keys in the output dictionary of the structured logging and will be used in the upcoming monitoring. If the variable is not defined in the configuration file, then it is taken from the name of the file, e.g. the job\_classification.jsonnet config file gives a default "channel\_name" value of "job\_classification".

**Note:** Added requirement on SQLAlchemy (for new datasource backend). Non-SQLAlchemy users should ensure the indexes from 13c2f283 are in their database.

**Note:** Added requirement on prometheus-client. Prometheus will be used as optional monitoring component.

**Note:** The "channel\_name" key in the Source Proxy config dictionaries needs to be changed to "source\_channel". "channel\_name" is now being used to describe the name of the channel itself, not the name of the channel the Source Proxy is gettting information from.

## Issues fixed in this release

- 481: Channel name should be available to all worker types in TaskManager
- 456: Logic engine messages show in the main DE log (1.6.99 post4) prj\_testing
- 458: Exception in new SQLAlchemy data source 1.6.99post4
- 455: New postgresql exception in 1.6.99post4 (aka Fixed databese inconsistency silently ignored in v1.6)
- 456: Logic engine messages show in the main DE log (1.6.99 post4)
- 451: Transforms executed in wrong order in 1.6.99.post3
- 367: Test race conditions bug
- 406: Taskmanager doesn't use/honor global log level
- 379: Add postgresql.sql to distributed decisionengine rpm
- 329: Docker container is missing pylint
- 293: Drop requirements.txt setup mode
- 285: Unify ProcessingState with Reaper state management code
- 253: Decision engine can sometimes start up at boot time before network name resolution is working (ae04db5)

#### Full list of commits since version 1.6.0

```
f42558df: Updated documentation for 1.7.0 release 029d118a: Updated release notes for 1.7.0 RC4 (1.6.99.post8) 0e19c754: fix SP 810994af: Update release_notes_1.7.rst
```

fbee95e7: Update release\_notes\_1.7.rst 68b955b0: Make sure product is a string

ef7a8b96: Automatically adjust PYTHONPATH for tests

e292d388: Updated release notes for 1.7.0 RC3 (1.6.99.post7)

d60b6e4e: new changes for logging with common logger name "channel"

8cdeb67e: Simplify return expression

8fb128d3: Ensure file is "flushed" so name is fully established

7806aa00: Add github CodeQL analysis

9f09bca9: removed modules/LogicEngine.py and corresponding test

b9d28fbf: Cleaner check for Any

cc91aa24: Switch to fstring formatting

7bb5b64f: Just return created value rather than store then return

f4847fbe: Combine nested with blocks

4ba38bcd: Drop redundant brackets

bdcfe8c9: By convention, pandas is usually imported as pd

1dd904ff: Use more traditional expression order

```
cccd31bc: Unused loop vars should start with _
c055a5cd: Drop _keys in favor of DB backed keys
e8c689b4: Moved prometheus-client requirement to proper place in list
5391500d: Added metrics API module
c2d7835c: Drop unnecessary timeout
c167fc50: Add tests for de-query-tool entry point
efabfeb3: Updated release notes for 1.7.0 RC2 (1.6.99.post6)
b2739c14: moved logging of LogicEngine from decisionengine logger to channel loggers
0c0532f3: Add locks to help ensure data changes are "atomic"
ae63c6ee: Use DB generated known keys so it always matches DB state
b2259e9e: Use public .keys() rather than internal implementation
85b6c3ba: Real world data shows the defaults are fine
95fb3fdf: Further constrain tablespace
3ebe8619: Finish implementation of get_datablock
edbb3568: Add entry point for de-query-tool
fed95c62: adding logging of importlib imports of modules
53e62f03: Sometimes pypy times out on the cleanup.
a44d4bc4: Don't test sqlite on pypy it isn't necessary
b13aa8a9: Some corrections
94c14110: Fix missing defines
5f102095: More detailed testing of datablock
b6c99021: Make sure our sqlite tests have ForeignKeyConditional support
6b76ba7c: Fix typo
6694369d: Ensure dbutils uses transactions
1df400ae: Fix spaces
5278fd99: Raise timout for numpy on pypy
6d0a1a74: Release notes ready for v1.7.0
084f74e1: Initial SQLAlchemy Datasource
3353aa00: Make sure our jsonnet is json synatx valid
402b1c26: Fix transform-ordering problem.
49297573: Fix incorrect packaging of tests at top level
fbfae499: The test_channel loads data once per second.
33f9ade1: Rename taskmanager test nodb
308343e9: Initial modifications for addition of structured logging
6f337b75: Add missing error message
```

23a4b770: Call fixtures in a cleaner manner for xdist

```
1f2fe8c4: Add self.config so I can introspect the fixtures later
689c0020: Add missing config attrib test
d2732816: Best practices are for fixtues to yield vs return
accef50a: Seed SQLAlchemy fixtures for later activation
31002bc5: Help define the fixture interlocking
0f5fb129: The pandas 1.3.0 doesn't build against PyPy any longer
a7d18a41: Correctly test datablock construction paths
9af4c144: the mock package was a backport for python2.
5ddaff8f: Add another constructor test
9ae9ad13: Make sure if the client says to stop we don't override it
a581cd2b: run pyupgrade against codebase for python3.6
09e4e79c: Handle reaper duplicate shutdowns more cleanly
64d29dc5: Drop pointless cache restore
1c6b2588: Update PyPy to 3.7 for testing
2bae173e: Increase wait for overloaded test workers, update log messages
b67c185c: When aborting CI builds cleanup all processes
6c5d6306: Trim pytest fast functions, add required plugin
8c63ca6b: note why we're ignoring this line
2bd4ecbc: Add a syntax check for the toml files
e2dca404: Sometimes these get stuck
6d012fab: Add in Jenkinsfile pipeline configuration a timeout at stage level
baf07973: Add timeout option to block-while/until
970faf92: Make pre-commit happy
0cea2285: Fix alignment issue
5620c65b: List why we aren't checking
88611d90: Ensure fixtures are cleaned up between invocations
0ba135d2: Setup blank DB for SQLAlchemy tests and prep fixtures
3793e674: Setup pre-commit
9e6d1317: Migrate test_Reaper to pytest fixtures
51df43bf: Cleanup a bunch of pointless whitespace
96e5d069: Fix typo
9f96f418: Setup datablock to use our paramaterized fixture
36ebc66c: Add config for LGTM
c6032e5f: Use topologically sorted transforms to remove some multi-threading.
e063f82a: Drop pointless comma
```

bfd6689e: Begin prepwork for PEP517

```
72c5725f: Stub out null source rather than more complex mocking
3b65e5e2: Push Singleton into its own space
fb5b177e: Put fixtures in central location
5ab3cbaa: Add more details to channel startup logs
afe7f7d7: Add log about what DB we are hitting
38034b2c: Let the datasource handle the connections internally
5e03b6fe: Since we are opening an IPv4 socket, just use 127.0.0.1 to check
cac2bef3: Fix missing version requirements
3be8f84f: Add line lenght for autoformater
90e2baad: Protect against inappropriate wait under error condition.
943a17a7: Fix de-client typo and adjust tests accordingly.
3b104eba: Set the logs to DEBUG for testing
4c5564d4: Add another sync method to try and make tests less spotty
66bd81f2: Make sure to encourage updates to tools
d16f04cc: Put postgresql datasource schema into RPM
62b97e79: Fix str so it includes all the data
611ef1f8: Drop pointless lines
5b9e2fb6: Drop unreachable excepts
6991f65f: Restore product-name translation required for some source-proxy cases.
f6258c09: Fixed formatting and updated content
104a0446: Update index.rst
2ed61289: Update index.rst
cb687150: Create release_notes.rst
3b57d4a2: Note new requirement
871af08b: Added 1.7.0 release notes
ce42b802: improved 1.6 release note
583c10fb: fixed rst error
96d4dc1e: Added 1.6.2 release notes, from branch 1.6
13c2f283: Add some helpful indexes to our default schema
29c32571: Log as workers are started
619021c2: One of these tests seems to be spotty, break them out to find which one
29a2c72d: Run the test in a way that gives us colors
4e36bfd2: Drop unused table create logic
5511f69e: Stronger notify state for when we've a lot of watchers.
b6cc7a46: Test the dataspace abstractions
```

e3b1f594: Better messages about our state

# decisionengine, Release 1.7.1.dev34+ge95071fd

2d2feab9: Drop duplicate tests, leave specifics 8e737329: Add parameter based datasource api tests 5c023aa5: Don't do debug logs for flake8, they aren't helpful f5d1a12f: Setup list of public exports for dataspace.py 7158b422: Merge pull request #365 from jcpunk/bad-update-is-error cd98cc4a: Update should error out if you try to do it wrongly eb7907fe: Add option to set taskmanager datestamp and sample usage e124532c: Make sure the fixture uses the production flow a8241b6e: Make sure RPM also owns the .egg-info so we don't confuse the namespaces da87376e: Ensure the DE server is fully started before running query 622bfacf: Simplify use of our PG fixtures df98ecdf: Fixed flake8 issue 061ff6cf: decisionengine/framework: stop\_channel runs Publisher shutdown methods 3727b80b: Fixup comment to avoid assuming this test uses the DB d45aaf6b: Fix script path typo a25a4a30: Fix ABC to match our actual usage 1510b2d1: Address minor linting issues 945e4b16: Fix missing attribute insert 5eace9d5: Add note for how to get modules in place 50a8e268: Add list of packages in the CI env to output b9cb197d: Sanity check the home directory cd17223c: Have client provide a hint when you ask for no behavior 95b02365: Fix de-query-tool to support produce/consume model e660ca72: Update required versions for bugfixes 6863cb81: Fix path error bb52e8b1: Merge pull request #340 from jcpunk/service-stop 6d7aba95: Drop obsolete files 168ae7aa: Name the tests better 0f60c4e3: Support new produces/consumes/configuration-description infrastructure. 81912469: Add de-query-tool 2a26c944: ExecStopPre is not supported on all systemd instances 67a54d5c: Merge pull request #338 from jcpunk/fix-pytest-postgres 70ab133f: Fixup use of pytest\_postgresql for version 3.0.0 f8f4255e: Merge pull request #337 from jcpunk/thread-names 5f49a4f6: Set names for the various parallel code 64da77c6: Merge pull request #327 from jcpunk/datablock-expire

```
de33a60a: Merge pull request #336 from knoepfel/use-toposort
31a8a905: Merge pull request #328 from knoepfel/de-class-inference
410e383d: Merge pull request #331 from jcpunk/reaper-interval-tests
719ff0c8: Test datablock expire funtions
e14c49d8: The 'name' parameter is optional.
7846c9f3: Enable DE class inference based on configuration.
32ab7e44: Use third-party topological sort.
01aa8ae6: Merge pull request #325 from jcpunk/channel-tests
52b48479: Merge pull request #326 from jcpunk/valid-config-tests
8c4749e7: Merge pull request #330 from jcpunk/pylint-actions
a37770c9: Ensure validation testing is tested
d8ab5eb6: Add missing test to ensure the run interval is actually used
0cd9c42b: Also run pylint for extra sanity checks
c5cf1fff: Ensure our errors error out
baf01700: Merge pull request #324 from jcpunk/cleanup-trivial-tests
2a0133aa: Try to cleanup trivial missing coverage
44e0ad6f: Merge pull request #323 from jcpunk/about-coverage
d811f617: Merge pull request #322 from knoepfel/fix-fail-on-error
cb426262: Merge pull request #312 from jcpunk/finish-setuptools
8f6d407d: Merge pull request #316 from jcpunk/abc-coverage
4d0676bb: Merge pull request #317 from vitodb/pylint
d7c43b96: Use regular expression to support fail_on_error feature.
ada66925: add support to run pylint tests
efb1e57b: Finish migration to pure setuptools
bc4720cf: We aren't testing 'unversioned' releases
e4dc35e3: Merge pull request #314 from jcpunk/jsonnet_syntax
87e32c22: Merge pull request #294 from jcpunk/move-reaper
dec85d5e: Merge pull request #319 from jcpunk/task-loop
4108472a: Merge pull request #320 from jcpunk/container-swig
920af1c9: Merge pull request #321 from knoepfel/include-init-files
650dffa7: Don't forget __init__.py files.
1b412e03: The latest m2crypto seems to need swig now
a6e3ab1c: Merge pull request #313 from jcpunk/conf-test
1205636a: Simplify run loop
```

30e59dc9: fix test client with no server verbose unit test for Jenkins CI (#315)

10384a8c: Move reaper into its own place and reuse state logic

# decisionengine, Release 1.7.1.dev34+ge95071fd

```
940584e4: No real way to test abstract base classes
250c14b1: The _validate function doesn't permit missing 'PRODUCES'
5ae1ce9f: Make sure syntax error in config names the problem
b899fa23: Add SourceProxy module test. (#307)
7b3df14c: Increae coverage of utils (#304)
ddba2a31: Fix duplicate entry warning (#311)
915673fa: Test modules minimally (#298)
bc0c21a9: Some repos may error out, don't let them kill the build (#297)
924a7047: doc: add 1.6.1 release notes
b1ab4d31: doc: fix typo
85e5d714: postgresql: do not print stack trace for low level library (#309)
255c6415: Setuptools uses entry return value as an error msg (#303)
2fd8db45: Fix name to match expectations (#305)
9cddb70a: updated release notes
7fe0358e: Error in more clean methods (#300)
84aa506c: Fix a bug in setup.py parsing of requirements. (#301)
a58b61bb: fix typo in release notes
```

# 1.1.2 Release 1.6.2

Patch level (bug fix) release.

#### Issues fixed in this release

#### Bugs fixed

- DEM 200 (part of it): Invoke correctly channels shutdown: (75eaa90)
- no issue: Use regular expression to support fail\_on\_error feature (1386d20)

#### Enhancements:

- Improved CI support (e.g. added pylint tests)
- 217: Add option to de-client –print-product to only print the column names in a data block and-or to print one or more records in key/value format. (c4c7681)

#### Full list of commits since version 1.6.1

ddba2a3: Fix duplicate entry warning (#311)

```
c4c7681: Updated de-query-tool w/ cherry pick of fixes from latest version of PR#332
f964d4b: Fixup use of pytest_postgresql for version 3.0.0
635ffd1: Also run pylint for extra sanity checks
11676ff: Fixed function w/ the same name
b8278f6: Add de-query-tool
75eaa90: Merge pull request #335 from shreyb/publisher_shutdown_from_1.6
77e3d79: Added set to shutdown method to TaskManager and accompanying test
1386d20: Merge branch 'knoepfel-fix-fail-on-error' into 1.6
73a18b1: Merge branch 'fix-fail-on-error' of https://github.com/knoepfel/decisionengine into knoepfel-fix-fail-on-
error
4f49fb7: Merge branch 'jcpunk-finish-setuptools' into 1.6
a5e5d39: Merge branch 'finish-setuptools' of https://github.com/jcpunk/decisionengine into jcpunk-finish-setuptools
a1ed252: Merge branch 'vitodb-pylint' into 1.6
c8eddda: Merge branch 'pylint' of https://github.com/vitodb/decisionengine into vitodb-pylint Meerging PR#317 to
release branch 1.6
d7c43b9: Use regular expression to support fail_on_error feature.
ada6692: add support to run pylint tests
efb1e57: Finish migration to pure setuptools
e4dc35e: Merge pull request #314 from jcpunk/jsonnet syntax
87e32c2: Merge pull request #294 from jcpunk/move-reaper
dec85d5: Merge pull request #319 from jcpunk/task-loop
4108472: Merge pull request #320 from jcpunk/container-swig
920af1c: Merge pull request #321 from knoepfel/include-init-files
650dffa: Don't forget init .py files.
1b412e0: The latest m2crypto seems to need swig now
a6e3ab1: Merge pull request #313 from jcpunk/conf-test
1205636: Simplify run loop
de553a7: fix test_client_with_no_server_verbose unit test for Jenkins CI (#315)
30e59dc: fix test_client_with_no_server_verbose unit test for Jenkins CI (#315)
10384a8: Move reaper into its own place and reuse state logic
250c14b: The _validate function doesn't permit missing 'PRODUCES'
5ae1ce9: Make sure syntax error in config names the problem
b899fa2: Add SourceProxy module test. (#307)
7b3df14: Increae coverage of utils (#304)
```

# decisionengine, Release 1.7.1.dev34+ge95071fd

```
915673f: Test modules minimally (#298)
bc0c21a: Some repos may error out, don't let them kill the build (#297)
924a704: doc: add 1.6.1 release notes
b1ab4d3: doc: fix typo
85e5d71: postgresql: do not print stack trace for low level library (#309)
255c641: Setuptools uses entry return value as an error msg (#303)
2fd8db4: Fix name to match expectations (#305)
9cddb70: updated release notes
7fe0358: Error in more clean methods (#300)
84aa506: Fix a bug in setup.py parsing of requirements. (#301)
a58b61b: fix typo in release notes
33660bf: fixed a typo[locuser@fermicloud462 decisionengine]
```

# 1.1.3 Release 1.6.1

Patch level (bug fix) release.

#### Issues fixed in this release

- 306 : /etc/decisionengine/decision\_engine.conf as shipped in RPM is wrong format (de0aef3)
- 275 : Running de-client –stop-channel <channel > results in KeyError (59fb44e)

## Full list of commits since version 1.6.0

```
d7ccd8a: doc: fix typo
ac48e50: updated release notes
de0aef3: Fix name to match expectations (#305)
59fb44e: postgresql: do not print stack trace for low level library (#309) (#310)
2162bbe: Setuptools uses entry return value as an error msg (#308)
b0fd9fb: 1.6.0 package backports (#302)
```

# 1.1.4 Release 1.6.0

## In this release:

- The logic engine has been rewritten in pure python. This removes the last C++ dependency the decision engine had. The build system has been updated accordingly.
- Migrated to setuptools package development library. This build system is the standard vanilla python build system
  provided with the python distribution. Build configurations have been updated and rpm packaging remains the
  primary distribution method.
- Completed logging implementation.

- Improvements in error handling and code coverage.
- Improvements in Jenkins and GitHub actions CI/CD pipelines.

#### Issues fixed in this release

- 44 : Logic Engine doesn't handle missing values gracefully (743effc)
- 253: Decision engine can sometimes start up at boot time before network name resolution is working (ae04db5)

# Full list of commits since version 1.5.0

```
2551e07: More coverage for de-client (#296)
dde3945: Make sure actions either complete in time or die (#295)
381861c: Update Jenkins pipeline configuration (#292)
eb771f4: Try to cleanup Dockerfile PATH issue (#291)
780cb56: fix unittest doc
8680942 : update unittest documentation
8154b24 : Fixup sphinx doc (#290)
5f7e13a: enhancements in logging and error handling in dataspace dir (#283)
3d92725 : Add missing runtime requirement (#286)
743effc: Allow conversion from errors to false values in logic-engine expressions. (#284)
124dcab: Inherit version from setuptools_scm if possible (#287)
3669803: added missing "" as line continuation
761f1d9: Drop invalid init.py
dc0e71b: migrate to setuptools (#264)
3b6f1bf: Make reaper reset state when starting from stopped proc (#280)
b2f9061: added ISO-8601 format to time in logging. changed name of function for better clarity. (#279)
0a74fe1: Improved DE client usage (#281)
ebf53e3: Added shutdown method to Publisher class (#278)
f95ab6d: Address some flake8/black reports (#274)
1c383b7: Automatically pull in our settings from about.py (#273)
e71f186: logging and error handling enhancements to taskmanager directory (#277)
7de9ab9: Increase Reaper log verbosity (#267)
019d245: Update actions to follow new best practices (#272)
b84e847 : Avoid possible sync issues in reaper startup (#271)
891975f: Remove vestigial C++ files. (#270)
42e5e1f: enhancements in logging and exception handling in newly added logicengine files (#265)
38effe6: Ensure the scheduler has started the thread before returning (#269)
db54fa1 : Start testing on PyPy with psycopg2cffi (#223)
```

# decisionengine, Release 1.7.1.dev34+ge95071fd

cc44058 : Squashed commit of the following: (#263)

d6548e9: Enhanced logging in the logicengine directory files (#261)

c341bf7: Better match our workflow with codecov (#260)

1fbe44d: Use 'new' syntax for forward compat (#259)

2294b0b : Do a limited pin on version requirements (#256)

bcda470: Python implementation of logic engine (#246)

c6721b4: address comment on RB

ae04db5: Add Wants and After (network-online.target) dependency

1a96b14 : Fix action repodata

a70cee8: Move to CodeCov.io

7b16b4e: Add Wants and Requires dependencies (#258)

76c3670 : Move to CodeCov.io (#254)

e7ba013: Fix action repodata (#255)

d7e72f2: revert 3.9 test

b04154b: added 1.5.0 release notes

a03da29: remove 3.9 to see if documentatoin gets generated

# 1.1.5 Release 1.5.0

#### In this release:

- Introduce data product query interface
- Cleanup of Ligic Engine code
- Improvements in error handling
- · Improvements in testing and CI

#### Issues fixed in this release

- 217, 218: Add option to de-client –print-product to only print the column names in a data block and-or to print one or more records in key/value format (fe7abcf)
- 240 : Logic Engine call leads to immediate taskmanager segfault exit (d855aa0)
- 239 : implement data product browsing interface (fe9faa9)

#### Full list of commits since version 1.4.1

```
d66c54b: Add PEP-0396 metadata (#243)
bfc91a6 : More compat between psycopg2/psycopg2cffi (#248)
f5d31a6: Cleanup Fixture FIXME (#249)
Odfaf3c: Adding docker documentation (#251)
4b166a2 : Since we are python3 only now, drop python-six compat layer (#252)
fe7abcf: Add format support to de-client (#217) (#241)
df5a3d7: Add wheel support for easier testing (#247)
7de970d : Add place to inject env if need be (#242)
84e2930 : Fix race in test case (#250)
d855aa0: Fix fact-lookup to support duplicate names in separate rules. (#245)
51370fb: Resolve fixture 'quickstart' issue (#238)
3ea9129: Move from TravisCI to raw actions (#235)
fe9faa9: implement data product browsing interface (#239)
cf0f3c0: Add support to use custom base docker container to run tests (#234)
d91722f: Compat with psycopg2cffi (#233)
7d15a8c: Test failing source proxy. (#232)
b9a4bbb: Add debug logs for which threads are created #176 (#231)
6e6f4c9: Updated Jenkins configuration documentation (#229)
2d9fd7b: Log if config passed validation #117 (#230)
60c46d3: Self-test needs a real namespace to 'import numpy' in new python eval (#228)
a120077: Test that the doc actually builds during CI (#227)
4b6240a: Extend timeout for coverage combine (#226)
b059696: Update workflow per changes at github (#225)
7a71cac : Use newer compilers/runtimes (#224)
15ffd93: Add header for strict includes (#222)
71b141a: Add special PyPy only requirement (#221)
9dbb932: Move Python C extension to versioned .so file (#220)
ea7ade5: Migrate from boost-python to pybind11 (#215)
e6b2eae: Add python 3.9 to testing matrix (#219)
04c8f9c: Add the option to print columns types on de-client (#216)
8815dc6 : Logic-engine cleanups (#211)
086d0d5: fix missing back tick
54cc084: modified release notes
24744cf: Synchronize access to the task managers (#214)
87a7fda: replde dash with underscore
```

# decisionengine, Release 1.7.1.dev34+ge95071fd

743d0fd: try sphinx\_rtd\_theme

18c7909: added 1.4.0 release notes

ff3d491: force docker pull when building the docker container to make sure to use an updated base layer (#210)

# 1.1.6 Release 1.4.1

#### In this release:

• Bug fixes to 1.4.0 release

#### Issues fixed in this release

• 213 : de-client hangs under certain circumstances in version 1.4 and greater (race condition) (84ecfe2)

#### Full list of commits since version 1.4.0

9799b9a: update release version to 1.4.1

84ecfe2: Synchronize access to the task managers (#214)

751b6b8 : Address data races; remove need to sleep in unit tests (#205)

# 1.1.7 Release 1.4.0

## In this release:

- Improvements in error handling and client/server interactions
- Added log rotation by time
- Improvements in code coverage

# Issues fixed in this release

- 153: Have de-client –print-product return different error message if product does not exist (18a950c)
- 171: yum update on decision engine rpm from python2 to python3 doesn't undo the symlinks (eb85c97)
- 188 : Channel debug info now leaks into startup.log (99d20a5)
- 208: Error when trying to run reaper in version 1.4.0 (84eccf3)

#### Full list of commits since version 1.3

84eccf3: Fix typo in reaper script. (#209)

d836abf: next RC

926944a: Fix coveralls reporting (#198) b95c323: Updating base Dockerfile (#199)

d302e31: Help jsonnet, which doesn't understand PosixPath objects. (#204)

2d791a7: Test configuration policies. (#197)

```
236e27a: Ensure items are returned in a stable order (#202)
e974f5f: add pylinit and pycodestyle (#203)
fbe7616: Test task manager (#196)
686ca80 : require more recent version of pytest-postgresql (#195)
99d20a5 : Fix double-logging problem. (#192)
4ce3d17: A set of fixtures to simplify unit tests (#183)
65f8052 : Fix typo (#190)
f3a4be8: Protect against None workers (#187)
ec310fb: remove py3 from package name
7006489: bump version to 1.4.0rc
158d835 : decisionengine/framework/modules: Fix SourceProxy retries (#184)
1356bf1 : Add support to test any branch in Jenkins (#182)
692fa8e : Add timeout support for unit test on Jenkins (#181)
e3d6e6a: Updated Jenkins documentation to take into account unit tests timeout parametr (#180)
2586a3e: Configuration redesign (#168)
fac984d: Fix error with DBUtils import. Looks like names of modules changed (#175)
7d661ee: Move postgres-specific implementation to postgres source. (#174)
eb85c97: Rpm (#173)
10fe843: Adding log rotation by time (#170)
a8d239b: Various improvements. (#167)
d9b92ee: Ignore vim's *.swp files (#166)
d9f72ef: Fix call to shutdown_timeout (and add sample entry to config) (#165)
3161795 : Add drops for items using tables being dropped (#164)
77d186d : Show output of test runtimes in travis (#163)
81820a4 : Allow server to start with no channels. (#161)
49879a6 : DE server and client usability improvements (#160)
de91c4f: Add tests to default and override config (#158)
14df1f6: Use python fallthrough for options (#159)
ac64a92: Drop python 2.7 integration tests since we are python3 only (#157)
d963301: Update Jenkins pipeline to properly test closing PR (#156)
64248cb: Merge 'runtime' tests into running channel tests (#150)
065ad77 : Adding Jenkins pipeline documentation (#155)
18a950c: fix print-product to report non-existing product as such (#154)
6493735 : Fix invalid attribute name (#152)
d953c6a: Remove unnecessary set start method call (#149)
c8c9b65 : guarantee that process is killed so test never hang (#147)
```

```
f1542b6: Channel test (#146)

7f349a8: Fix faulty TaskManager state type (#145)

d50f1c4: fix logging regression introduced in f5e299969e0611e3480e9fa2782052df... (#142)

becfa26: Pass the correct type. (#144)

1a60daf: DecisionEngine: fix typo (#143)

9e7b867: Updating Jenkins pipeline configuration (#140)

e3a6703: fix regression introduced in f5e299969e0611e3480e9fa2782052df86d7c4ed (#141)

4900bc6: Restore runtime test. (#139)

0823f3d: Consolidate DE server/client tests into one file. (#138)

4f84435: A few more access fixes.

160cfd1: Fix task manager state access.

c00d819: A few more cleanups.

ec087e2: Various cleanups

a309ffe: Improvements to DE client CLI.
```

## 1.1.8 Release 1.3.0

#### In this release:

- Introduced Jsonnet based configuration system
- · Improved logging
- · Improved coverage of datasource

# Full list of commits since version 1.2

```
239e82c: postgresql: improve SQL query (#133)
668eb1f: Update to make the code compatible with both python and JSON based config files (#129)
afd8837: Configuration-manager fixes (#128)
571e2be: Remove pip installed system python packages
407d9ed: Update Dockerfile
1fefc69: Implement unit tests for datablock.py (#122)
43c8d7a: Adjust global configuration to include program-option values. (#126)
2840813: Switch to Jsonnet configuration system (#125)
5c4ae0e: logging changes: added config file and command line interface (#124)
6697f22: Further config-manager testing and factorizations. (#123)
fa89fd0: Insulate multiprocessing test from parent environment. (#120)
139a537: Allow empty base directory for log file. (#119)
f14d40c: Factorize configuration-loading steps. (#118)
e00afee: Enhance testing and error reporting of ConfigManager (#117)
```

```
c3d1be3: Python 3 upgrades. (#116)
e7399af: Header fix (#114)
0456abf: Adding editor config file, see https://editorconfig.org/ (#115)
82112d1: Dockerfile: fetch osg 3.5 repo rpm (#113)
97c21b1: osg version 3.5 (#112)
33f28a8: Introduce jsonnet dependency (#110)
3f8b55e: improve server error handling (#108)
f15588e: added 1.2.0 release notes
b433325: Remove unnecessary 'main' functionality. (#107)
```

## 1.1.9 Release 1.2.0

In this release:

- Swithed to python3
- · Improved coverage
- · Database data retention: added reaper to remove data older than configurable number of days
- · Improved logging

# decisionengine

```
3dfe167: Jenkins pipeline improvements (#106)
22a7073: pull request for review request 137 (#105)
cafffb2: Make it possible to run code directly (for tests), and (#100)
802e98b: replace psycog2 witt psycopg2-binary (#101)
573ce8f: Jenkins pipeline improvements (#99)
9d08835 : Run coveralls even under failed state (#97)
bc1df4b: Add tests for PostgreSQL datasource (#71)
c1ac391: Fix missing py-modules.html (#96)
8dbfdee: Setup gh-pages doc workflow (#94)
cd4a01a : Doc (#93)
673080d: set version to 1.2.0 (for now). Supply conf file that corresponds to (#91)
f912225: Db (#92)
dc8b68a: Add reaper to the RPC (#83) (#90)
29ade91: adding .Jenkinsfile with Jenkins pipeline configuration (#86)
c1dfe5c: Don't exclude E1004 from pylint, do exclude line breaks (#89)
440f949 : Fix varname (#88)
313d135 : Compress (#87)
6b8dc4b: Revert "Add reaper to the RPC (#83)"
```

```
dbea8e5: Update utils.sh so pytest will complete.
e848316: Update to postgresq111
7f4b805: Add reaper to the RPC (#83)
0ba2c51: remove astpp module and depedencies it pulls in (#81)
6b8eab9: don't track test coverage of tests (#80)
Oda18ec: made reaper.py executable
aca24a3: make reaper.py executable, make symbolic link to it from /usr/bin (#72)
0202acf: Implementation of data reaper (#70)
16b6be1: Simple changes for Python 3 deployment (#69)
fd2418c: Fix warnings caught by PEP-8 Speaks.
d16359b: Python 3 (and other) simplications.
3c7b6b7 : Only run Github Actions for python3.6 (#68)
453cbba: Update README.md
b27ed53: remove unnecessary (and atually harmful) python shebang (#66)
decisionengine modules
30d928b: clone version 1.2.0 of decisionengine
ae7c5a6: Jenkins pipeline improvements (#236)
310befd: T198 (#235)
a65886d: Fix import as reported in: https://github.com/HEPCloud/decisionengin... (#232)
93711cc: Run coveralls even if tests fail (#229)
03d763a: Jenkins pipeline improvements (#230)
f48d30f: Fix/223 (#228)
c8aa262 : github ticket 199 (#222)
0323bda: Address: https://github.com/HEPCloud/decisionengine_modules/issues/224 (#226)
62e4df6: Add support to run CI on Jenkins (#221)
5ab1541 : bump master version to 1.2.0 (for now) (#219)
bc19c65: decisionengine_modules/NERSC: Added retry loop for NERSC API Calls (#220)
41a50de : Sync up pep8speaks and run_pylint.sh with decisionengine settings (#218)
db4634f : silence pylint error (#217)
1b95141: Fix whitespace around operator error
746ea38: ignore W503
8a8b5f4: remove unused variable
a6668bf: fix PEP8 warnings
13773ee: address pep8 warnings
6bea4ca: silence pylint error
```

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

a1d0507: fixing pep8 warning

a10bd17: debugging one import error ec501ad: make coveralls.io links work

deab1a7 : T201 (#204) 69f2645 : Add coveragerc

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

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

# 1.1.10 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/
31/
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

# **CHAPTER**

# **TWO**

# **DEVELOPER DOCUMENTATION**

The developer documentation is in the GitHub Wiki

Intructions to build the package, or to run unit tests and other CI tests, and to install decisionengine are in the GitHub Wiki as well.

**CHAPTER** 

**THREE** 

# **JENKINS CI PIPELINE**

# 3.1 Decisionengine CI with Jenkins pipeline

Jenkins dashboard with Decisionengine framework CI results is available here.

A CI build is triggered any time a PR is created/closed or a commit is made to an existing PR. There are also *nightly CI builds* to test a list of predefined branches.

The Jenkins pipeline runs *pylint* and *unit\_tests* test suites alongside the *rpmbuild* stage.

The Jenkins dashboard looks like this:



On the bottom left side there is the list of recent CI builds that are named after the PR or the branch tested. On the bottom right side the dashboard shows for each CI build detailed status for each test suite.

Hovering the mouse over the status box for each CI build stage, a tool-tip with a button to access log details shows up.

Next to the build number the symbol <sup>1</sup> gives access to a menu with the list of artifacts stored for that build. Those artifacts include logs and the tarball with RPMs.

From the panel on the left side it is possible to access the PR on GitHub by clicking on the PR icon that looks like this  $\frac{1142}{112}$ .

On occasion it could be useful to trigger a manual CI build to test a branch on the official DE GitHub repository or on

the user fork. For this purpose, on the top left panel the user can click on the Build with Parameters and this panel shows up

button.

# Pipeline decisionengine\_pipeline

This build requires parameters:



the user can modify these parameters to customize what code to test with the CI build.

The *DE\_REPO* parameter can point to the user fork or to the main repository.

The BRANCH parameter can point to the desired branch to test.

The PYTEST\_TIMEOUT parameter is the timeout in seconds for unit\_tests.

When ready, by clicking on the Build button, the CI build will start.

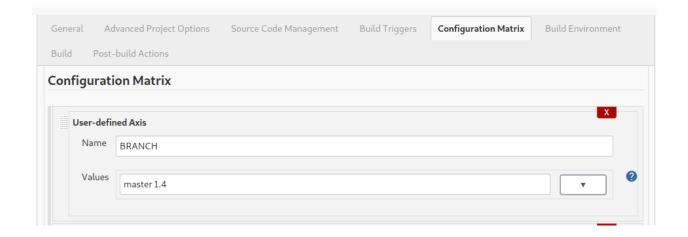
The pipeline configuration is part of the decisionengine repo.

# 3.1.1 Nightly CI build configuration

The nightly CI build for Decisionengine framework uses this Jenkins project that triggers a CI build using the Jenkins pipeline described above to test a list of predefined branches.



Branches to test are defined using the project matrix as shown in the picture below. Each branch in the list (here *master* and *1.4*) spawns an independent CI build.



In the *Build* section of the configuration it is set the list of Jenkins subprojects to be triggered, in this case we have *decisionengine\_pipeline* and *decisionengine\_modules\_pipeline*.

The *Parameters* text box is used to override parameters of each Jenkins subproject with a custom value. In total this Jenkins project triggers 4 CI builds, i.e. 2 branches X 2 Jenkins subprojects.



Finally the *Build Triggers* section is used to setup the schedule for the periodic build, in this case it is scheduled to run at about 2 AM.

Jenkins will choose the actual time depending on the actual load on the system.



**CHAPTER** 

# **FOUR**

# **SOURCE CODE**

# 4.1 Welcome to decisionengine's documentation!

# 4.1.1 decisionengine package

**Subpackages** 

decisionengine.framework package

**Subpackages** 

decisionengine.framework.config package

**Subpackages** 

decisionengine.framework.config.tests package

# **Submodules**

# decisionengine.framework.config.tests.test\_config module

```
decisionengine.framework.config.tests.test_config._channel_config_dir(relative_dir)

decisionengine.framework.config.tests.test_config._global_config_file(relative_filename)

decisionengine.framework.config.tests.test_config.load()

decisionengine.framework.config.tests.test_config.test_channel_empty_config(load, caplog)

decisionengine.framework.config.tests.test_config.test_channel_empty_dictionary(load, caplog)

decisionengine.framework.config.tests.test_config.test_channel_invalid_modules_list(load, caplog)

decisionengine.framework.config.tests.test_config.test_channel_invalid_modules_no_keys(load, caplog)

decisionengine.framework.config.tests.test_config.test_channel_invalid_modules_string(load, caplog)

decisionengine.framework.config.tests.test_config.test_channel_loading(caplog)
```

```
decisionengine.framework.config.tests.test_config.test_channel_module_missing_all(load,
                                                                                    caplog)
decisionengine.framework.config.tests.test_config.test_channel_module_missing_module(load,
decisionengine.framework.config.tests.test_config.test_channel_module_missing_parameters(load,
                                                                                           caplog)
decisionengine.framework.config.tests.test_config.test_channel_names(load)
decisionengine.framework.config.tests.test_config.test_channel_no_config_files(load)
decisionengine.framework.config.tests.test_config.test_channel_no_modules(load)
decisionengine.framework.config.tests.test_config.test_empty_config(load)
decisionengine.framework.config.tests.test_config.test_minimal_jsonnet_right_extension(load,
                                                                                         sys)
decisionengine.framework.config.tests.test_config.test_minimal_jsonnet_wrong_extension(load,
                                                                                         cap-
                                                                                         sys)
decisionengine.framework.config.tests.test_config.test_syntax_error_in_config_names_bad_file(load)
decisionengine.framework.config.tests.test_config.test_valid_but_empty_config(load)
decisionengine.framework.config.tests.test de std module
decisionengine.framework.config.tests.test_de_std.config(basename, jpathdirs=None)
decisionengine.framework.config.tests.test_de_std.test_allow_duplicate_source_proxy_keys()
decisionengine.framework.config.tests.test_de_std.test_combine_one_level()
decisionengine.framework.config.tests.test_de_std.test_combine_one_level_skip_proxies()
decisionengine.framework.config.tests.test_de_std.test_error_on_duplicate_keys()
decisionengine.framework.config.tests.test_de_std.test_jpath()
decisionengine.framework.config.tests.test policies module
decisionengine.framework.config.tests.test_policies.test_channel_config_dir(tmp path,
                                                                             monkeypatch)
decisionengine.framework.config.tests.test_policies.test_global_config_dir(tmp_path,
                                                                            monkeypatch)
decisionengine.framework.config.tests.test_policies.test_global_config_file(tmp_path,
                                                                             monkeypatch)
decisionengine.framework.config.tests.test_policies.test_valid_dir(tmp_path)
```

### **Module contents**

### **Submodules**

# decisionengine.framework.config.ChannelConfigHandler module

Manager of channel configurations.

The ChannelConfigHandler manages only channel configurations and not the global decision-engine configuration. It is responsible for loading channel configuration files and validating that the channels have the correct configuration artifacts.

nel\_config\_dir)

```
Bases: object
_load_channel(channel_name, path)
get_channels()
```

# load\_all\_channels()

Load all channel configurations inside the stored channel-configuration directory.

Any cached configurations will be dropped prior to reloading.

# load\_channel(channel\_name)

Load a single configuration for a channel with the supplied name.

The behavior is to read a configuration file whose path is:

```
<cached channel config. dir>/{channel_name}.jsonnet
```

where the cached channel-configuration directory was stored whenever the ChannelConfigHandler object was created, and {channel\_name} is the value of the supplied method argument.

```
print_channel_config(channel)
```

decisionengine.framework.config.ChannelConfigHandler.\_check\_keys(channel\_conf\_dict) check that channel config has mandatory keys:type data: dict

decisionengine.framework.config.ChannelConfigHandler.\_make\_de\_logger(global\_config)

# decisionengine.framework.config.ValidConfig module

ValidConfig represents a valid JSON document.

The decision engine requires each of its configuration files to be valid JSON. This is achieved by either supplying a valid Jsonnet or JSON document upfront.

Vetting of a file for JSON validity happens upon construction of a 'ValidConfig' object. A fully constructed 'Valid-Config' object thus corresponds to a valid JSON document.

ValidConfig represents a valid JSON configuration in the form of a dictionary.

In addition to the normal dictionary operations, users may call 'dump()' to print out in a string form the JSON configuration.

```
_abc_impl = <_abc._abc_data object>
dump()
```

Print dictionary data to a valid JSON string.

decisionengine.framework.config.ValidConfig.\_config\_from\_file(config\_file, jpaths=None)

# decisionengine.framework.config.policies module

Decision-engine default configuration policies.

For the decision-engine process, the configuration policies are:

- The global configuration file must be named 'decision\_engine.jsonnet' and it must reside in (a) a directory that can be accessed through the 'CONFIG\_PATH' environment variable, or (b) the /etc/decisionengine directory.
- All channel configurations must reside in (a) a directory accessible through the 'CHANNEL\_CONFIG\_PATH' environment variable, or (b) a 'config.d' subdirectory of the /etc/decisionengine directory.

The utilities provided in this module provide simple means of accessing the configuration artifacts according to the policies listed above. Please consult the documentation for each function below for more detailed information.

```
decisionengine.framework.config.policies.channel_config_dir(parent_dir=None)
```

Retrieve the channel configuration directory as a pathlib.Path object.

This function returns a path object according to the following precedence rules:

- 1. If the 'parent\_dir' argument is provided, the returned path object will correspond to '{parent\_dir}/config.d'.
- 2. If the 'CHANNEL\_CONFIG\_PATH' environment variable has been set, the returned path object will correspond to \${CHANNEL\_CONFIG\_PATH}.
- 3. If neither 1 or 2 apply, the returned path object corresponds to '{global\_config\_dir()}/config.d' (see documentation for 'global\_config\_dir()').

Regardless of the precedence rule used, the returned path object must be a valid directory or an exception will be raised—i.e. if the 'parent\_dir' argument is supplied, and the resulting path object is not a valid directory, the function will exit with an exception and not attempt rule 2 or 3.

```
decisionengine.framework.config.policies.global_config_dir()
```

Retrieve global configuration dir as pathlib.Path object.

This is the directory that houses the 'decision\_engine.jsonnet' global configuration file.

This function checks that the 'CONFIG\_PATH' variable has been set or will use /etc/decisionengine otherwise. If the path exists as a directory, then the directory path is returned as a string; otherwise an exception is raised.

```
decisionengine.framework.config.policies.global_config_file(parent dir=None)
```

Return the pathlib.Path object corresponding to the global configuration.

If supplied, the 'parent\_dir' is assumed to be the full path corresponding to a directory containing the 'decision\_engine.jsonnet' file. If not provided, the global configuration directory is determined based on the behavior of the 'global\_config\_dir()' function.

An exception is raised if no 'decision\_engine.jsonnet' file is found.

```
decisionengine.framework.config.policies.valid_dir(path, scope)
```

Throws if the supplied path object is not a directory, otherwise returns the path object.

### **Module contents**

decisionengine.framework.dataspace package

**Subpackages** 

decisionengine.framework.dataspace.datasources package

**Subpackages** 

decisionengine.framework.dataspace.datasources.sqlalchemy\_ds package

**Submodules** 

decisionengine.framework.dataspace.datasources.sqlalchemy ds.datasource api module

The datasource layer for our abstraction

```
class decisionengine.framework.dataspace.datasources.sqlalchemy_ds.datasource_api.SQLAlchemyDS(config_dic
     Bases: decisionengine.framework.dataspace.datasource.DataSource
     A DecisionEngine data source via the SQL Alchemy ORM
     {
          "dataspace": {
                                            "decisionengine.framework.dataspace.datasources.sqlalchemy_ds",
              "datasource": { "module":
                  "name": "SQLAlchemyDS", "params": {
                    "pool_size": 5, "max_overflow": 10, "timeout": 30,
                                                                                                 "di-
                    # url is mandatory, but any engine keyword is accepted here.
                                                                                         "url":
                    alect[+driver]://user:password@host/dbname"
              }
          }
     }
     Exceptions should be caught and logged by the caller.
     _abc_impl = <_abc._abc_data object>
     close()
          Close all connections to the database
              Returns None
     connect()
          Create a pool of database connections
              Returns None
     create tables()
          Create database tables
```

Returns None

### delete\_data\_older\_than(days)

Delete data older that interval

Parameters days (int) – remove data older than this many days

Returns None

### 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 (str/uuid) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to clone
- **new\_generation\_id** (int) generation id to create

### Returns None

# get\_datablock(taskmanager\_id, generation\_id)

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

#### **Parameters**

- $taskmanager_id(str/uuid)$  id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to locate

**Returns** with all set keys and their associated values

Return type dict

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

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

### **Parameters**

- $taskmanager_id(str/uuid)$  id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to locate
- **key** (str) key for the value

**Returns** The possibly binary value stored earlier

Return type obj

# get\_dataproducts(taskmanager\_id, key=None)

Return list of all data products associated with with taskmanager\_id

### **Parameters**

- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- **key** (*str*) key for the value

**Returns** each element is the matching row as a dict()

### Return type tuple

### 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 (str/uuid) – id of taskmanager to retrieve

- **generation\_id** (*int*) generation id to locate
- **key** (*str*) key for the value

### Returns

**fields in order are:** taskmanager.taskmanager\_id, header.taskmanager\_id, header.generation\_id, header.key, header.create\_time, header.scheduled create time, header.creator, header.schema id

### Return type tuple

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

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

### **Parameters**

- **taskmanager\_name** (*str*) name of taskmanager to retrieve
- taskmanager\_id (str/uuid) id of taskmanager to retrieve

**Returns** the largest generation stored within the database

## Return type int

# 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 (str/uuid) id of taskmanager to retrieve
- $generation_id(int)$  generation id to locate
- **key** (*str*) key for the value

## Returns

**fields in order are:** taskmanager.taskmanager\_id, metadata.taskmanager\_id, metadata.generation\_id, metadata.key, metadata.state, metadata.generation\_time, metadata.missed\_update\_count

# Return type tuple

# get\_schema(table=None)

Given the table name return it's schema

# get\_taskmanager(taskmanager\_name, taskmanager\_id=None)

Find the task manager by name/uuid in the database get back the primary key.

If multiples match, find highest primary key.

### **Parameters**

- $taskmanager_name (str)$  name of taskmanager to retrieve
- $taskmanager\_id(str/uuid)$  id of taskmanager to retrieve

**Returns** the matching row, column names as keys

### Return type dict

# get\_taskmanagers(taskmanager\_name=None, start\_time=None, end\_time=None)

Find taskmanagers that meet our search

# **Parameters**

•  $taskmanager_name(str)$  – name of taskmanager to retrieve

- **start\_time** (*datetime*) Datetime to confine against
- end\_time (datetime) Datetime to confine against

Returns each element is a dict() matching row, column names as keys

# Return type list

**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 (str/uuid) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to create
- **key** (*str*) key for the value
- value (obj) Value can be an object or dict or a binary
- header (datablock.Header) Header for the value
- metadata (datablock.Metadata) Metadata for the value

#### **Returns** None

### reset\_connections()

Reset the connection to the database. So long as self.engine isn't undef, the engine can still make new connections if new db actions happen. It just wont have any open at this time.

#### Returns None

store\_taskmanager(name, taskmanager\_id, datestamp=None)

Store TaskManager in database

## **Parameters**

- name (str) name of taskmanager to retrieve
- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- datestamp (datetime) datetime of created object, defaults to 'now'

**Returns** the primary key of the row in the database

# Return type int

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** (*str/uuid*) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to update
- $\mathbf{key}$  ( $\mathbf{str}$ ) key for the value
- value (obj) Value can be an object or dict or a binary
- header (datablock.Header) Header for the value
- metadata (datablock.Metadata) Metadata for the value

### Returns None

# decisionengine.framework.dataspace.datasources.sglalchemy ds.db schema module

The table layout and utilities for our SOLAlchemy ORM class decisionengine.framework.dataspace.datasources.sqlalchemy\_ds.db\_schema.Base(\*\*kwargs) Bases: object The base class of the class hierarchy. When called, it accepts no arguments and returns a new featureless instance that has no instance attributes and cannot be given any. \_sa\_registry = <sqlalchemy.orm.decl\_api.registry object> metadata = MetaData() registry = <sqlalchemy.orm.decl\_api.registry object> class decisionengine.framework.dataspace.datasources.sqlalchemy\_ds.db\_schema.Dataproduct(\*\*kwargs) Bases: sqlalchemy.orm.decl\_api.Base The PRIMARY KEY on this table isn't used.... Existing code appears to depend on column order. \_sa\_class\_manager = {'generation\_id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'key': <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager': <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager\_id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'value': <sqlalchemy.orm.attributes.InstrumentedAttribute object>} generation\_id id kev taskmanager taskmanager\_id value class decisionengine.framework.dataspace.datasources.sqlalchemy\_ds.db\_schema.Header(\*\*kwargs) Bases: sqlalchemy.orm.decl\_api.Base The PRIMARY KEY on this table isn't used.... The existing code has a hard expectation on the time columns being BIGINT rather than datetime objects burried within the classes.

Looks like there was an inital goal of a relationship with the Schema table, but it may not be in use

Existing code appears to depend on column order.

```
_sa_class_manager = {'create_time':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'creator':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'expiration_time':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'generation_id':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'id':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'key':
     <sglalchemv.orm.attributes.InstrumentedAttribute object>. 'scheduled create time';
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'schema_id':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager_id':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>}
    create_time
    creator
    expiration_time
    generation_id
    id
    key
    scheduled_create_time
     schema_id
    taskmanager
    taskmanager_id
class decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_schema.Metadata(**kwargs)
    Bases: sqlalchemy.orm.decl_api.Base
    The PRIMARY KEY on this table isn't used....
    The existing code has a hard expectation on the state field as a 'text' element.
    The existing code has a hard expectation on the time columns being BIGINT rather than datetime objects burried
    within the classes.
    Existing code appears to depend on column order.
     _sa_class_manager = {'generation_id':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'generation_time':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'id':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'key':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'missed_update_count':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'state':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager':
     <sqlalchemy.orm.attributes.InstrumentedAttribute object>, 'taskmanager_id':
    <sqlalchemy.orm.attributes.InstrumentedAttribute object>}
    generation_id
    generation_time
    id
    kev
    missed_update_count
    state
```

```
taskmanager
     taskmanager_id
class decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_schema.Schema(**kwargs)
     Bases: sqlalchemy.orm.decl_api.Base
     This table may not be in use
     Has a one-to-many relationship with: Header - may not be in use
     _sa_class_manager = {'schema': <sqlalchemy.orm.attributes.InstrumentedAttribute
     object>, 'schema_id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>}
     schema
     schema id
class decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_schema.Taskmanager(**kwargs)
     Bases: sqlalchemy.orm.decl_api.Base
     Has a one-to-many relationship with: Header Metadata Dataproduct
     changes cascade on: Header Metadata Dataproduct
     Existing code appears to depend on column order.
     _sa_class_manager = {'datestamp': <sqlalchemy.orm.attributes.InstrumentedAttribute
     object>, 'name': <sqlalchemy.orm.attributes.InstrumentedAttribute object>,
     'sequence_id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>,
     'task_dataproduct': <sqlalchemy.orm.attributes.InstrumentedAttribute object>,
     'task_header': <sqlalchemy.orm.attributes.InstrumentedAttribute object>,
     'task_metadata': <sqlalchemy.orm.attributes.InstrumentedAttribute object>,
     'taskmanager_id': <sqlalchemy.orm.attributes.InstrumentedAttribute object>}
     datestamp
     name
     sequence_id
     task dataproduct
     task header
     task_metadata
     taskmanager_id
decisionengine.framework.dataspace.datasources.sqlalchemy_ds.utils module
Code not written by us
decisionengine.framework.dataspace.datasources.sglalchemy_ds.utils.add_engine_pidguard(engine)
     Based on https://stackoverflow.com/questions/62920507/using-sqlalchemy-connection-pooling-queues-with-python-multiprocess
decisionengine.framework.dataspace.datasources.sqlalchemy_ds.utils.clone_model(model,
                                                                                    **kwargs)
     Based on https://stackoverflow.com/a/55991358
decisionengine.framework.dataspace.datasources.sqlalchemy_ds.utils.orm_as_dict(obj)
     Based on: https://stackoverflow.com/a/37350445
```

### **Module contents**

```
Top level import so we can rationally segment items of the ORM
```

```
class decisionengine.framework.dataspace.datasources.sqlalchemy_ds.SQLAlchemyDS(config_dict)
     Bases: decisionengine.framework.dataspace.datasource.DataSource
     A DecisionEngine data source via the SQL Alchemy ORM
          "dataspace": {
              "datasource": { "module":
                                             "decisionengine.framework.dataspace.datasources.sqlalchemy_ds",
                  "name": "SQLAlchemyDS", "params": {
                    "pool size": 5, "max overflow": 10, "timeout": 30,
                    # url is mandatory, but any engine keyword is accepted here.
                                                                                           "url":
                                                                                                    "di-
                    alect[+driver]://user:password@host/dbname"
                  }
              }
          }
     Exceptions should be caught and logged by the caller.
     _abc_impl = <_abc._abc_data object>
          Close all connections to the database
              Returns None
     connect()
          Create a pool of database connections
              Returns None
     create_tables()
          Create database tables
              Returns None
     delete_data_older_than(days)
          Delete data older that interval
              Parameters days (int) – remove data older than this many days
              Returns None
     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 (str/uuid) – id of taskmanager to retrieve
                   • generation_id (int) – generation id to clone
                   • new_generation_id (int) - generation id to create
              Returns None
```

### get\_datablock(taskmanager id, generation id)

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

#### **Parameters**

- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to locate

**Returns** with all set keys and their associated values

# Return type dict

# 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 (str/uuid) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to locate
- $\mathbf{key}$  ( $\mathbf{str}$ ) key for the value

**Returns** The possibly binary value stored earlier

# Return type obj

### get\_dataproducts(taskmanager\_id, key=None)

Return list of all data products associated with with taskmanager\_id

#### **Parameters**

- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- **key** (*str*) key for the value

**Returns** each element is the matching row as a dict()

## Return type tuple

# 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 (str/uuid) id of taskmanager to retrieve
- $generation_id(int)$  generation id to locate
- **key** (*str*) key for the value

### **Returns**

```
fields in order are: taskmanager.taskmanager_id, header.taskmanager_id, header.generation_id, header.key, header.create_time, header.scheduled_create_time, header.creator, header.schema_id header.expiration_time,
```

### Return type tuple

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

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

- $taskmanager_name (str)$  name of taskmanager to retrieve
- taskmanager\_id (str/uuid) id of taskmanager to retrieve

**Returns** the largest generation stored within the database

### Return type int

### 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 (str/uuid) id of taskmanager to retrieve
- **generation\_id** (int) generation id to locate
- **key** (*str*) key for the value

#### Returns

**fields in order are:** taskmanager.taskmanager\_id, metadata.taskmanager\_id, metadata.generation\_time, metadata.missed\_update\_count metadata.state, metadata.generation\_time, metadata.missed\_update\_count

# Return type tuple

### get\_schema(table=None)

Given the table name return it's schema

### get\_taskmanager(taskmanager\_name, taskmanager\_id=None)

Find the task manager by name/uuid in the database get back the primary key.

If multiples match, find highest primary key.

#### **Parameters**

- **taskmanager\_name** (*str*) name of taskmanager to retrieve
- taskmanager\_id (str/uuid) id of taskmanager to retrieve

**Returns** the matching row, column names as keys

### Return type dict

get\_taskmanagers(taskmanager\_name=None, start\_time=None, end\_time=None)

Find taskmanagers that meet our search

# **Parameters**

- $taskmanager_name(str)$  name of taskmanager to retrieve
- **start\_time** (*datetime*) Datetime to confine against
- end\_time (datetime) Datetime to confine against

**Returns** each element is a dict() matching row, column names as keys

# Return type list

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

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

- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- **generation\_id** (*int*) generation id to create
- **key** (*str*) key for the value
- value (obj) Value can be an object or dict or a binary

- header (datablock.Header) Header for the value
- metadata (datablock.Metadata) Metadata for the value

### Returns None

### reset\_connections()

Reset the connection to the database. So long as self.engine isn't undef, the engine can still make new connections if new db actions happen. It just wont have any open at this time.

#### Returns None

store\_taskmanager(name, taskmanager\_id, datestamp=None)

Store TaskManager in database

#### **Parameters**

- name (str) name of taskmanager to retrieve
- taskmanager\_id (str/uuid) id of taskmanager to retrieve
- datestamp (datetime) datetime of created object, defaults to 'now'

**Returns** the primary key of the row in the database

# Return type int

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 (str/uuid) id of taskmanager to retrieve
- **generation\_id** (int) generation id to update
- **key** (*str*) key for the value
- value (obj) Value can be an object or dict or a binary
- header (datablock.Header) Header for the value
- metadata (datablock.Metadata) Metadata for the value

Returns None

### decisionengine.framework.dataspace.datasources.tests package

# **Submodules**

### decisionengine.framework.dataspace.datasources.tests.fixtures module

pytest fixtures/constants

decisionengine.framework.dataspace.datasources.tests.fixtures.PG\_DB\_WITHOUT\_SCHEMA(request:

\_pytest.fixtures.Fixture

 $\rightarrow$ 

psycopg2.extensions.c

Fixture factory for PostgreSQL.

Parameters request - fixture request object

Returns postgresql client

decisionengine.framework.dataspace.datasources.tests.fixtures.**PG\_DE\_DB\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_*. Load our PG schema into the database via this fixture so pytest knows the limitations on parallel usage of this database scope.

decisionengine.framework.dataspace.datasources.tests.fixtures.PG\_PROG(request:

\_pytest.fixtures.FixtureRequest, tmpdir\_factory: \_pytest.tmpdir.TempdirFactory) →

Iterator[pytest\_postgresql.executor.PostgreS

Process fixture for PostgreSQL.

Parameters request – fixture request object

Returns tcp executor

- decisionengine.framework.dataspace.datasources.tests.fixtures.**SQLALCHEMY\_PG\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITH* Get a blank database from pytest\_postgresql. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.
- decisionengine.framework.dataspace.datasources.tests.fixtures.SQLALCHEMY\_TEMPFILE\_SQLITE(tmp\_path) Setup an SQLite database with the pytest tmp\_path fixture. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.
- decisionengine.framework.dataspace.datasources.tests.fixtures.datasource(request)
  This parameterized fixture will setup up various datasources.

Add datasource objects to DATASOURCES\_TO\_TEST once they've got our basic schema loaded. And adjust our *if* statements here until we are SQLAlchemy only.

Pytest should take it from there and automatically run it through all the tests using this fixture.

decisionengine.framework.dataspace.datasources.tests.fixtures.mock\_data\_block()
This fixture replaces the standard datablock implementation.

The current DataBlock implementation does not own any data products but forwards them immediately to a backend datasource. The only implemented datasource requires Postgres, which is overkill when needing to test simple data-product communication between modules.

This mock datablock class directly owns the data products, thus avoiding the need for a datasource backend. It is anticipated that a future design of the DataBlock will own the data products, thus making this mock class unnecessary.

# decisionengine.framework.dataspace.datasources.tests.test datasource api module

This test plan covers a generic dataspace object via pytest parameters.

- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_create\_tables(datasource) create\_tables() should be safe to call multiple times
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_delete\_data\_older\_than\_are Can we delete old entries
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_duplicate\_datablock(dataso Can we duplicate taskmanager1 and all its entries
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_datablock(datasource)
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_dataproduct(datasource)

  Can we get the dataproduct by uuid with key

- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_dataproduct\_not\_exist Does it error out if we ask for bogus information?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_dataproducts(datasource Can we get the dataproducts by uuid and uuid with key
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_dataproducts\_not\_exis Does it error out if we ask for bogus information?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_header(datasource)

  Can we fetch a header?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_header\_not\_exist(datas Does it error out if we ask for a bogus header?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_last\_generation\_id(dataset\_datasource\_api.test\_get\_last\_generation\_id(dataset\_datase

decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_last\_generation\_id\_no

- Does it error out if we ask for a bogus taskmanager?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_metadata(datasource)

  Can we fetch a metadata element?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_metadata\_not\_exist(datasource Does it error out if we ask for a bogus metadata element?
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_taskmanager\_exists(datasource\_api test\_get\_taskmanager\_exists).
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_taskmanager\_not\_exist
  This should error out
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_taskmanagers(datasourceCan I get multimple task managers

decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_get\_taskmanagers\_not\_exis

- Do I error out when asking for garbage
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_has\_config(datasource)

  This should have a config dict we can pass to jsonnet
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_insert(datasource)

  Can we insert new elements
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_reset\_connections(datasource reset\_connections() should be safe to call any time

 $\tt decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_store\_taskmanager({\it datasource}) and {\it datasource} and {\it data$ 

- Can we make new entries

  decisionengine framework dataspace datasources tests test datasource ani test undate(datasource)
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_update(datasource)

  Do updates work as expected
- decisionengine.framework.dataspace.datasources.tests.test\_datasource\_api.test\_update\_bad(datasource)

  Do updates fail to work on bogus taskmanager as expected

# decisionengine.framework.dataspace.datasources.tests.test postgresql module

decisionengine.framework.dataspace.datasources.tests.test\_postgresql.test\_generate\_insert\_query()

### Module contents

### **Submodules**

### decisionengine.framework.dataspace.datasources.null module

# class decisionengine.framework.dataspace.datasources.null.NullDataSource(config\_dict)

Bases: decisionengine.framework.dataspace.datasource.DataSource

Implementation of data source ABC that does nothing

```
_abc_impl = <_abc._abc_data object>
```

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 interval :type days: long :arg days: remove data older than 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\_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

 ${\tt get\_dataproduct}(\textit{taskmanager\_id}, \textit{generation\_id}, \textit{key})$ 

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

- 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\_dataproducts(taskmanager id, key=None)

Return list of all data products associated with with taskmanager\_id

Parameters key (string) – data product key

# 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

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

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

#### **Parameters**

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

# 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

# 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

# get\_taskmanagers(taskmanager\_name=None, start\_time=None, end\_time=None)

Retrieve TaskManagers :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

### reset\_connections()

Drop any cached connections and reconnect to the database

### store\_taskmanager(name, taskmanager\_id, datestamp=None)

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 :type datestamp: datetime :arg datestamp: datetime of created object, defaults to 'now'

# 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.datasources.postgresql module

```
class decisionengine.framework.dataspace.datasources.postgresql.Postgresql(config dict)
     Bases: decisionengine.framework.dataspace.datasource.DataSource
     Implementation of postgresql data source
     __query(query_string, values=None, cursor_factory=None)
     _abc_impl = <_abc._abc_data 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\_dataproducts(taskmanager\_id, key=None)

Return list of all data products associated with with taskmanager\_id

Parameters key (string) – data product key

### 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

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

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

### **Parameters**

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

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

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

- 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

### get\_taskmanagers(taskmanager name=None, start time=None, end time=None)

Retrieve TaskManagers :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

### reset\_connections()

Drop any cached connections and reconnect to the database

### store\_taskmanager(name, taskmanager\_id, datestamp=None)

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 :type datestamp: datetime :arg datestamp: datetime of created object, defaults to 'now'

```
tables = {'dataproduct': ['taskmanager_id TEXT', 'generation_id INT', 'key TEXT',
'value BLOB'], 'header': ['taskmanager_id TEXT', 'generation_id INT', 'key TEXT',
'create_time REAL', 'expiration_time REAL', 'scheduled_create_time REAL', 'creator
TEXT', 'schema_id INT'], 'metadata': ['taskmanager_id TEXT', 'generation_id INT',
'key TEXT', 'state TEXT', 'generation_time REAL', 'missed_update_count INT'],
'schema': ['schema_id INT', 'schema BLOB']}
```

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

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

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

# decisionengine.framework.dataspace.tests package

### **Submodules**

# decisionengine.framework.dataspace.tests.fixtures module

decisionengine.framework.dataspace.tests.fixtures.PG\_DB\_WITHOUT\_SCHEMA(request:

\_pytest.fixtures.FixtureRequest)

 $\rightarrow$ 

psycopg2.extensions.connection

Fixture factory for PostgreSQL.

Parameters request - fixture request object

Returns postgresql client

decisionengine.framework.dataspace.tests.fixtures.**PG\_DE\_DB\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_SCHEMA*) Load our PG schema into the database via this fixture so pytest knows the limitations on parallel usage of this database scope.

 $\label{lem:continuous} {\tt decisionengine.framework.dataspace.tests.fixtures.PG\_PROG(\it request:\_pytest.fixtures.FixtureRequest, tmpdir\_factory:$ 

 $\_pytest.tmpdir.TempdirFactory) \rightarrow$ 

Iterator[pytest\_postgresql.executor.PostgreSQLExecutor]

Process fixture for PostgreSQL.

Parameters request - fixture request object

Returns tcp executor

decisionengine.framework.dataspace.tests.fixtures.**SQLALCHEMY\_PG\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_SCHEMA*) Get a blank database from pytest\_postgresql. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

decisionengine.framework.dataspace.tests.fixtures.**SQLALCHEMY\_TEMPFILE\_SQLITE**(*tmp\_path*) Setup an SQLite database with the pytest tmp\_path fixture. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

decisionengine.framework.dataspace.tests.fixtures.datasource(request)

This parameterized fixture will setup up various datasources.

Add datasource objects to DATASOURCES\_TO\_TEST once they've got our basic schema loaded. And adjust our *if* statements here until we are SQLAlchemy only.

Pytest should take it from there and automatically run it through all the tests using this fixture.

decisionengine.framework.dataspace.tests.fixtures.dataspace(request)

This parameterized fixture will setup up various datasources. Add datasource objects to DATA-SOURCES\_TO\_TEST once they've got our basic schema loaded. And adjust our *if* statements here until we are SQLAlchemy only.

Pytest should take it from there and automatically run it through all the tests using this fixture.

decisionengine.framework.dataspace.tests.fixtures.load\_sample\_data\_into\_datasource(schema\_only\_db) load our sample test data into a dataspace This is a function not a fixture so you can run it on any datasource providing the right API.

# decisionengine.framework.dataspace.tests.test\_Reaper module

```
decisionengine.framework.dataspace.tests.test_Reaper.config()
decisionengine.framework.dataspace.tests.test_Reaper.reaper(request)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_bad_config(reaper, config)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_missing_config(reaper,
                                                                               config)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_missing_config_key(reaper,
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_small_retain(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_small_run_interval(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_start_two_reapers(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_fail_wrong_config_key(reaper,
decisionengine.framework.dataspace.tests.test_Reaper.test_just_stop_no_error(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_loop_of_start_stop_in_clumps(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_reap_default_state(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_reaper_can_reap(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_source_fail_can_be_fixed(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_start_delay(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_start_stop(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_start_stop_stop(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_state_can_be_active(reaper)
decisionengine.framework.dataspace.tests.test_Reaper.test_state_sets_timer_and_uses_it(reaper)
```

# decisionengine.framework.dataspace.tests.test datablock module

```
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_constructor(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_duplicate(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_get_dataproducts(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_get_header(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_get_metadata(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_get_taskmanager(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_is_expired(dataspace)
     This test just validates the method/function exists. The stub within our default code should be replaced by a class
     inheriting from it. That class should have more rational return types.
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_is_expired_with_key(dataspace)
     This test just validates the method/function exists. The stub within our default code should be replaced by a class
     inheriting from it. That class should have more rational return types.
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_key_management(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_key_management_change_name(datasp
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_mark_expired(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_no_key_by_name(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_DataBlock_to_str(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_Header_constructor(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_Header_is_valid(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_Metadata_constructor(dataspace)
decisionengine.framework.dataspace.tests.test_datablock.test_Metadata_set_state(dataspace)
decisionengine.framework.dataspace.tests.test datablock zlib module
decisionengine.framework.dataspace.tests.test_datablock_zlib.test_compress()
decisionengine.framework.dataspace.tests.test_datablock_zlib.test_zdumps()
decisionengine.framework.dataspace.tests.test_datablock_zlib.test_zloads()
decisionengine.framework.dataspace.tests.test datasource module
```

decisionengine.framework.dataspace.tests.test\_datasource.test\_has\_methods\_we\_expect()

# decisionengine.framework.dataspace.tests.test dataspace module

- decisionengine.framework.dataspace.tests.test\_dataspace.test\_dataspace\_config\_finds\_bad()
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_duplicate\_datablock(dataspace)

  Can we duplicate taskmanager1 and all its entries
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_datablock(dataspace)

  Can we get the datablock content
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_dataproduct(dataspace)

  Can we get the dataproduct by uuid with key
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_dataproduct\_not\_exist(dataspace)

  Does it error out if we ask for bogus information?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_dataproducts(dataspace)

  Can we get the dataproducts by uuid and uuid with key
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_dataproducts\_not\_exist(dataspace)

  Does it error out if we ask for bogus information?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_header\_not\_exist(dataspace)

  Does it error out if we ask for a bogus header?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_last\_generation\_id(dataspace)

  Can we get the last generation id by name or name and uuid
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_last\_generation\_id\_not\_exist(dataspace)

  Does it error out if we ask for a bogus taskmanager?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_metadata(dataspace)

  Can we fetch a metadata element?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_metadata\_not\_exist(dataspace)

  Does it error out if we ask for a bogus metadata element?
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_taskmanager\_exists(dataspace)

  Can I get a taskmanager by name or name and uuid
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_taskmanager\_not\_exists(dataspace)
  This should error out
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_taskmanagers(dataspace)

  Can I get multimple task managers
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_get\_taskmanagers\_not\_exist(dataspace)

  Do I error out when asking for garbage
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_has\_config(dataspace) verify our config entry exists
- decisionengine.framework.dataspace.tests.test\_dataspace.test\_store\_taskmanager(dataspace)

  Can we make new entries

decisionengine.framework.dataspace.tests.test\_dataspace.test\_update\_bad(dataspace)

Do updates fail to work on bogus taskmanager as expected

### **Module contents**

### **Submodules**

# decisionengine.framework.dataspace.datablock module

\_setitem(key, value, header, metadata=None)
put a 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_dataproducts(key=None)
```

get\_header(key)

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(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 :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
class decisionengine.framework.dataspace.datablock.Header(taskmanager_id, create_time=None,
                                                                  expiration_time=None,
                                                                  scheduled create time=None,
                                                                  creator='module', schema id=None)
     Bases: collections.UserDict
     _abc_impl = <_abc._abc_data 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', 'schema_id', 'taskmanager_id'}
exception decisionengine.framework.dataspace.datablock.InvalidMetadataError
     Bases: Exception
     Errors due to invalid Metadata
class decisionengine.framework.dataspace.datablock.Metadata(taskmanager_id, state='NEW',
                                                                    generation id=None,
                                                                    generation time=None,
                                                                    missed update count=0)
     Bases: collections.UserDict
     _abc_impl = <_abc._abc_data object>
     required_keys = {'generation_id', 'generation_time', 'missed_update_count', 'state',
     'taskmanager_id'}
     set_state(state)
          Set the state for the Metadata
     valid_states = {'END_CYCLE', 'METADATA_UPDATE', 'NEW', 'START_BACKUP'}
```

class decisionengine.framework.dataspace.datablock.ProductRetriever(product\_name,

product\_type,
product source)

Bases: object

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\_impl = <\_abc.\_abc\_data 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

 $\textbf{abstract duplicate\_datablock}(\textit{taskmanager\_id}, \textit{generation\_id}, \textit{new\_generation\_id})$ 

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

- 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\_dataproducts(taskmanager\_id, key)

Return list of all data products associated with with taskmanager\_id

Parameters key (string) – data product key

# 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(taskmanager\_name, taskmanager\_id=None)

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

#### **Parameters**

- taskmanager\_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

# abstract get\_taskmanagers(taskmanager\_name=None, start\_time=None, end\_time=None)

Retrieve TaskManagers :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

- 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 reset\_connections()

Drop any cached connections and reconnect to the database

abstract store\_taskmanager(taskmanager\_name, taskmanager\_id, datestamp=None)

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 :type datestamp: datetime :arg datestamp: datetime of created object, defaults to 'now'

# 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

# ${\bf class} \ \ {\bf decisionengine.framework.dataspace.dataspace.DataSpace} ({\it config})$

Bases: object

DataSpace class is collection of datablocks and provides interface to the database used to store the actual data

```
close()
```

```
delete(taskmanager_id, all_generations=False)
```

duplicate\_datablock(taskmanager\_id, generation\_id, new\_generation\_id)

 ${\tt get\_datablock}(\textit{taskmanager\_id}, \textit{generation\_id})$ 

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

get\_dataproducts(taskmanager\_id, key=None)

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)
     get_taskmanagers(taskmanager_name=None, start_time=None, end_time=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, taskmanager_id, datestamp=None)
     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
\textbf{exception} \ \ \text{decisionengine.} framework. data space. data space. \textbf{DataSpaceError}
     Bases: Exception
     Errors related to database access
exception decisionengine.framework.dataspace.dataspace.DataSpaceExistsError
     Bases: Exception
     Errors related to database access
decisionengine.framework.dataspace.maintain module
class decisionengine.framework.dataspace.maintain.Reaper(config)
     Bases: object
     Reaper provides functionality of periodic deletion of data older than retention interval in days
     The class attributes indicate a rational set of defaults that shouldn't be altered by user configuration.
     MIN_RETENTION_INTERVAL_DAYS = 7
     MIN_SECONDS_BETWEEN_RUNS = 7080
     _reaper_loop(delay)
          The thread actually runs this.
     reap()
          Actually spawn the query to delete the old records. Lock the state as this task doesn't have a cancel option.
     property retention_interval
          We have data constraints, so use a property to track
     property seconds_between_runs
          We have data constraints, so use a property to track
     start(delay=0)
          Start thread with an optional delay to start the thread in X seconds
```

```
stop()
```

Try to stop the reaper, will block if the reaper cannot be interupted.

### **Module contents**

decisionengine.framework.engine package

# **Subpackages**

decisionengine.framework.engine.tests package

### **Submodules**

decisionengine.framework.engine.tests.fixtures module

pytest defaults

A DE Server using a private database

decisionengine.framework.engine.tests.fixtures.PG\_DB\_WITHOUT\_SCHEMA(request:

 $\_pytest.fixtures.FixtureRequest)$   $\rightarrow$ 

psycopg2.extensions.connection

Iterator[pytest\_postgresql.executor.PostgreSQLExecutor]

Fixture factory for PostgreSQL.

Parameters request - fixture request object

Returns postgresql client

decisionengine.framework.engine.tests.fixtures.**PG\_DE\_DB\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_SCHEMA*) Load our PG schema into the database via this fixture so pytest knows the limitations on parallel usage of this database scope.

decisionengine.framework.engine.tests.fixtures. $PG_PROG(request: \_pytest.fixtures.FixtureRequest, tmpdir\_factory: \_pytest.tmpdir.TempdirFactory) <math>\rightarrow$ 

Process fixture for PostgreSQL.

Parameters request – fixture request object

**Returns** top executor

decisionengine.framework.engine.tests.fixtures.**SQLALCHEMY\_PG\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_SCHEMA*)

Get a blank database from pytest\_postgresql. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

decisionengine.framework.engine.tests.fixtures.**SQLALCHEMY\_TEMPFILE\_SQLITE**(*tmp\_path*)

Setup an SQLite database with the pytest tmp\_path fixture. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

# decisionengine.framework.engine.tests.test\_client\_only module

```
decisionengine.framework.engine.tests.test_client_only.test_client_err_returned_as_rc()
    no de server is running, so –status should error
decisionengine.framework.engine.tests.test_client_only.test_client_err_returned_verbose_as_rc()
    no de server is running, so -status should error
decisionengine.framework.engine.tests.test_client_only.test_client_help(capfd)
decisionengine.framework.engine.tests.test_client_only.test_client_with_no_command_says_use_help()
decisionengine.framework.engine.tests.test_client_only.test_client_with_no_server()
decisionengine.framework.engine.tests.test_client_only.test_client_with_no_server_verbose()
decisionengine.framework.engine.tests.test_client_only.test_exclusive_options()
decisionengine.framework.engine.tests.test query tool only module
decisionengine.framework.engine.tests.test_query_tool_only.test_client_err_returned_as_rc()
    no de server is running, so –status should error
decisionengine.framework.engine.tests.test_query_tool_only.test_client_err_returned_verbose_as_rc()
    no de server is running, so –status should error
decisionengine.framework.engine.tests.test_query_tool_only.test_query_tool_help()
decisionengine.framework.engine.tests.test_query_tool_only.test_query_tool_with_no_server()
```

decisionengine.framework.engine.tests.test\_query\_tool\_only.test\_query\_tool\_with\_no\_server\_verbose()

### decisionengine.framework.engine.tests.test startup module

```
decisionengine.framework.engine.tests.test_startup._check_override(arguments)
decisionengine.framework.engine.tests.test_startup.test_change_port()
decisionengine.framework.engine.tests.test_startup.test_default_config()
```

#### **Module contents**

### **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.

```
{\bf class} \ \ {\bf decisionengine.framework.engine.DecisionEngine.DecisionEngine} ( {\it global\_config}, \\ {\it channel\_config\_loader}, \\ {\it server\_address}) \\ {\bf Bases: socketserver.ThreadingMixIn, xmlrpc.server.SimpleXMLRPCServer} \\ {\it \_dataframe\_to\_column\_names} ( {\it df}) \\ \\
```

```
_dataframe_to_csv(df)
_dataframe_to_json(df)
_dataframe_to_table(df)
_dataframe_to_vertical_tables(df)
_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.

```
block_until(state, timeout=None)
block_while(state, timeout=None)
get_logger()
handle_sighup(signum, frame)
reaper_start(delay)
reaper_status()
reaper_stop()
rm_channel(channel, maybe_timeout)
rpc_block_while(state_str, timeout=None)
rpc_get_channel_log_level(channel)
rpc_get_log_level()
rpc_kill_channel(channel, timeout=None)
rpc_print_product(product, columns=None, query=None, types=False, format=None)
rpc_print_products()
rpc_query_tool(product, format=None, start_time=None)
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_rm_channel(channel, maybe_timeout)
rpc_set_channel_log_level(channel, log_level)
    Assumes log_level is a string corresponding to the supported logging-module levels.
rpc_show_config(channel)
    Show the configuration for a channel.
```

```
rpc_show_de_config()
     rpc_start_channel(channel_name)
     rpc_start_channels()
     rpc_status()
     rpc_stop()
     rpc_stop_channel(channel)
     rpc_stop_channels()
     start_channel(channel_name, channel_config)
     start_channels()
     stop_channels()
     stop_worker(worker, timeout)
class decisionengine.framework.engine.DecisionEngine.RequestHandler(request, client_address,
     Bases: xmlrpc.server.SimpleXMLRPCRequestHandler
     rpc_paths = ('/RPC2',)
class decisionengine.framework.engine.DecisionEngine.StopState(value)
     Bases: enum. Enum
     An enumeration.
     Clean = 2
     NotFound = 1
     Terminated = 3
decisionengine.framework.engine.DecisionEngine._channel_preamble(name)
decisionengine.framework.engine.DecisionEngine._create_de_server(global_config,
                                                                        channel_config_loader)
     Create the DE server with the passed global configuration and config manager
decisionengine.framework.engine.DecisionEngine._get_de_conf_manager(global config dir,
                                                                           channel config dir,
                                                                           options)
decisionengine.framework.engine.DecisionEngine._get_global_config(config_file, options)
decisionengine.framework.engine.DecisionEngine._start_de_server(server)
     Start the DE server and listen forever
decisionengine.framework.engine.DecisionEngine.main(args=None)
     If args is None, sys.argv will be used instead If args is a list, it will be used instead of sys.argv (for unit testing)
decisionengine.framework.engine.DecisionEngine.parse_program_options(args=None)
     If args is a list, it will be used instead of sys.argv
```

### decisionengine.framework.engine.Workers module

Class that encapsulates a channel's task manager as a separate process.

This class' run function is called whenever the process is started. If the process is abruptly terminated—e.g. the run method is pre-empted by a signal or an os.\_exit(n) call—the Worker object will still exist even if the operating-system process no longer does.

To determine the exit code of this process, use the Worker.exitcode value, provided by the multiprocessing.Process base class.

This class manages and provides access to the task-manager workers.

The intention is that the decision engine never directly interacts with the workers but refers to them via a context manager:

```
with workers.access() as ws: # Access to ws now protected ws['new_channel'] = Worker(...)
```

In cases where the decision engine's block\_while or block\_until methods must be called (e.g. during tests), one should used the unguarded access:

```
with workers.unguarded_access() as ws: # Access to ws is unprotected ws['new_channel'].wait_until(...)
```

Calling a blocking method while using the protected context manager (i.e. workers.access()) will likely result in a deadlock.

### decisionengine.framework.engine.de\_client module

 ${\tt decisionengine.framework.engine.de\_client.console\_scripts\_main} ({\it args\_to\_parse=None})$ 

This is the entry point for the setuptools auto generated scripts. Setuptools thinks a return from this function is an error message.

decisionengine.framework.engine.de\_client.create\_parser()

decisionengine.framework.engine.de\_client.execute\_command\_from\_args(argsparsed, de\_socket) argsparsed should be from create\_parser in this file

decisionengine.framework.engine.de\_client.main(args\_to\_parse=None)

If you pass a list of args, they will be used instead of sys.argv

#### decisionengine.framework.engine.de query tool module

decisionengine.framework.engine.de\_query\_tool.console\_scripts\_main(args\_to\_parse=None)

This is the entry point for the setuptools auto generated scripts. Setuptools thinks a return from this function is an error message.

decisionengine.framework.engine.de\_query\_tool.create\_parser()

decisionengine.framework.engine.de\_query\_tool.execute\_command\_from\_args(argsparsed, de\_socket)

Calls the proper function for the arguments passed to de\_query\_tool.

#### **Parameters**

- **argsparsed** (*Namespace*) Should be from create\_parser in this file.
- **de\_socket** (*ServerProxy*) RPC Server Proxy.

**Returns** Output of the command.

Return type str

**Parameters args\_to\_parse** (*list*, *optional*) – If you pass a list of args, they will be used instead of sys.argv. Defaults to None.

**Returns** Query result

Return type str

#### Module contents

decisionengine.framework.logicengine package

#### **Subpackages**

decisionengine.framework.logicengine.tests package

# **Submodules**

### decisionengine.framework.logicengine.tests.test\_bool\_function\_name module

decisionengine.framework.logicengine.tests.test\_bool\_function\_name.test\_error\_conditions()

# decisionengine.framework.logicengine.tests.test\_cascaded\_rules module

```
decisionengine.framework.logicengine.tests.test_cascaded_rules.myengine()
decisionengine.framework.logicengine.tests.test_cascaded_rules.test_rule_that_does_not_fire(myengine)
decisionengine.framework.logicengine.tests.test_cascaded_rules.test_rule_that_fires(myengine)
```

#### decisionengine.framework.logicengine.tests.test construction module

```
decisionengine.framework.logicengine.tests.test_construction.test_configuration_with_fact_using_function
decisionengine.framework.logicengine.tests.test_construction.test_configuration_with_numy_facts()
decisionengine.framework.logicengine.tests.test_construction.test_default_construction()
    LogicEngine is not default constructible.
```

decisionengine.framework.logicengine.tests.test\_construction.test\_trivial\_configuration()

Logic engine constructed with trivial rules and facts.

decisionengine.framework.logicengine.tests.test\_construction.test\_wrong\_configuration()

LogicEngine construction requires rules and facts; if we don't supply them it is an error.

## decisionengine.framework.logicengine.tests.test\_duplicate\_fact\_names module

decisionengine.framework.logicengine.tests.test\_duplicate\_fact\_names.test\_duplicate\_fact\_names()

#### decisionengine.framework.logicengine.tests.test facts module

```
decisionengine.framework.logicengine.tests.test_facts.make_db(maximum)

decisionengine.framework.logicengine.tests.test_facts.test_compound_fact_with_spaces()

decisionengine.framework.logicengine.tests.test_facts.test_fact_using_numpy_array()

decisionengine.framework.logicengine.tests.test_facts.test_fact_using_numpy_function()

decisionengine.framework.logicengine.tests.test_facts.test_fact_with_fail_on_error()

decisionengine.framework.logicengine.tests.test_facts.test_fact_with_nested_names()

decisionengine.framework.logicengine.tests.test_facts.test_simple_fact()

decisionengine.framework.logicengine.tests.test_facts.test_syntax_error(caplog)
```

### decisionengine.framework.logicengine.tests.test\_fail\_on\_error module

```
decisionengine.framework.logicengine.tests.test_fail_on_error.logic_engine_with_fact(fact)
decisionengine.framework.logicengine.tests.test_fail_on_error.test_conditional_fact()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_fact_with_misspecified_attribute()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_fail_on_error(caplog)
decisionengine.framework.logicengine.tests.test_fail_on_error.test_false_fact_with_spaces()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_false_literal_fact()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_index_error()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_misspecified_fact()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_true_fact()
decisionengine.framework.logicengine.tests.test_fail_on_error.test_true_literal_fact()
decisionengine.framework.logicengine.tests.test pandas fact module
decisionengine.framework.logicengine.tests.test_pandas_fact.mydata(y)
    Return a 'datablock' surrogate carrying a Pandas DataFrame, and a parameter named 'y' with value y.
decisionengine.framework.logicengine.tests.test_pandas_fact.myengine()
decisionengine.framework.logicengine.tests.test_pandas_fact.test_rule_that_does_not_fire(myengine)
    Rules that do not fire do not create entries in the returned actions and newfacts.
decisionengine.framework.logicengine.tests.test_pandas_fact.test_rule_that_fires(myengine)
decisionengine.framework.logicengine.tests.test rule with negated fact module
decisionengine.framework.logicengine.tests.test_rule_with_negated_fact.myengine()
decisionengine.framework.logicengine.tests.test_rule_with_negated_fact.test_rule_that_does_not_fire(mye
    Rules that do not fire do not create entries in the returned actions and newfacts.
decisionengine.framework.logicengine.tests.test_rule_with_negated_fact.test_rule_that_fires(myengine)
decisionengine.framework.logicengine.tests.test_simple_configuration module
decisionengine.framework.logicengine.tests.test_simple_configuration.myengine()
decisionengine.framework.logicengine.tests.test_simple_configuration.test_error_on_bad_names(myengine)
decisionengine.framework.logicengine.tests.test_simple_configuration.test_rule_that_does_not_fire(myeng.
    Rules that do not fire do not create entries in the returned actions and newfacts.
decisionengine.framework.logicengine.tests.test_simple_configuration.test_rule_that_fires(myengine)
```

#### **Module contents**

#### **Submodules**

#### decisionengine.framework.logicengine.BooleanExpression module

## decisionengine.framework.logicengine.FactLookup module

Establishes a policy for looking up a fact based on the given name.

To wit, the first fact with a given name is the one that is used in the evaluation of all subsequent facts.

As an example, consider the following configuration:

```
facts: { should_publish: "(True)",
}, rules: {
    publish_1: { expression: "should_publish", facts: ["should_publish"]
    }, publish_2: {
        expression: "should_publish", actions: ["go_to_press"] facts: ["should_publish"]
    } retract: {
        expression: "not should_publish", facts: ["should_retract"]
}
```

In the above, the first fact to be evaluated will always be the top-level facts (i.e. those not encapsulated by the 'rules' table). The rules labeled 'publish\_1' and 'publish\_2' both rely on the 'should\_publish' fact in their expressions, and they in turn create their own facts with the same name. FactLookup ensures that 'publish\_1' and 'publish\_2' will both use the evaluated fact from the top-level 'facts' table.

```
rule_for(fact_name)
```

Selects rule required to evaluate fact with the supplied name.

```
Parameters fact_name (str) – Name of fact for which rule will be selected. 
Return type str
```

Returns Rule name

```
sorted_rules(rules_cfg)
```

Rules sorted according to rule dependencies.

```
Parameters rules_cfg (dict) - rules as specified in logic-engine configuration
```

Return type list

**Returns** Rules to be evaluated by the rule engine.

#### decisionengine.framework.logicengine.LogicEngine module

```
class decisionengine.framework.logicengine.LogicEngine.LogicEngine(cfg)
     Bases: decisionengine.framework.modules.Module.Module
     _create_facts_dataframe(newfacts)
          Convert newfacts dict in format below to dataframe with columns ['rule name', 'fact name', fact value']
          facts dict format: 'newfacts': {
               'publish_glidein_requests': { 'allow_hpc_new': True, 'allow_foo': True
              }, 'dummy_rule': {
                   'dummy new fact': True
          }
     consumes()
          Return the names of all the items that must be in the DataBlock for the rules to be evaluated.
     evaluate(db)
          Evaluate our facts and rules, in the context of the given data. db can be any mappable, in particular a
          DataBlock or dictionary.
              Parameters db (DataBlock) – Products used to evaluate facts.
     evaluate_facts(db)
              Parameters db (DataBlock) – Products used to evaluate facts.
              Return type dict
              Returns Evaluated fact values (e.g. True or False) for each fact name.
     produces()
decisionengine.framework.logicengine.LogicEngine.passthrough_configuration(publisher_names)
     Assembles logic-engine configuration to unconditionally execute all publishers.
decisionengine.framework.logicengine.Rule module
```

```
class decisionengine.framework.logicengine.Rule.Rule(rule_name, rule_cfg)
     Bases: object
     In-memory representation of logic-engine rule, relying on parsing utilities in BooleanExpression.
     evaluate(evaluated_facts)
          Evaluates a compiled expression given the supplied facts.
```

Parameters evaluated\_facts (dict) - Initial fact values (e.g. True or False) for each fact name.

**Return type** bool

### decisionengine.framework.logicengine.RuleEngine module

Engine responsible for evaluating logic-engine rules.

This class is responsible for (a) forming a sorted set of rules that supports dependencies between them, and (b) evaluating the rules according to a specified fact-lookup policy.

```
execute(evaluated facts)
```

Evaluates all rules given the supplied facts.

**Parameters evaluated\_facts** (dict) – Initial fact values (e.g. True or False) for each fact name.

Return type tuple

**Returns** Actions to be taken based on rule evaluation; new facts produced during that evaluation.

#### **Module contents**

# decisionengine.framework.managers package

#### **Submodules**

#### decisionengine.framework.managers.ChannelManager module

```
Channel Manager
```

Decision Channel. Instantiates runners according to channel configuration

**class** decisionengine.framework.managers.ChannelManager.ChannelManager(name, dataspace,

generation\_id, channel\_dict, global\_config, subscribe\_queue, channel\_subscribed)

 $Bases:\ decision engine.\ framework.\ managers.\ Component \texttt{Manager}.\ Component \texttt{Manager}.$ 

Channel Manager: Runs decision cycle for transforms and publishers

#### check\_for\_new\_data\_products()

Check the incoming data product queue, insert into datablock, and update source\_new\_data appropriately

#### decision\_cycle()

Decision cycle to be run periodically (by trigger)

# do\_backup()

Duplicate current data block and return its copy

Return type DataBlock

## register\_with\_sources(channel\_id, channel\_name, all\_sources)

Sends registration information for the sources that the channel is interested in to the SourceSubscription-Manager

```
reset_source_flags(sources)
     Sets self.source new data to False for the sources which are being used by this decision cycle
         Parameters sources (list) – list of source names that are to be set back to the "not-updated"
run()
     Channel 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_transform(transform, data_block)
     Run a transform
         Parameters
             • transform (TaskRunner) – source TaskRunner
             • data_block (DataBlock) - data block
run_transforms(data block=None)
     Run transforms. So far in main process.
         Parameters data_block (DataBlock) – data block
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_all_sources(channel sources)
     Wait for all sources this channel is interested in to finish their execution
         Parameters channel_sources (list) – list of sources that need to be polled for completion
wait_for_any_source(channel sources)
     Waits for any source this channel is interested in to post an updated data block before allowing continuation
         Parameters channel_sources (list) - list of sources to be watched for updated data
```

wait\_for\_registration(channel\_id)

**class** decisionengine.framework.managers.ChannelManager.**TaskRunner**(conf dict)

Bases: object

Provides interface to loadable modules and events for synchronization of execution

decisionengine.framework.managers.ChannelManager.\_make\_runners\_for(configs)

#### decisionengine.framework.managers.ComponentManager module

```
Decision Engine ComponentManager (Base class for ChannelManager and SourceManager)
class decisionengine.framework.managers.ComponentManager.ComponentManager(name,
                                                                                                                                                                                                         generation_id,
                                                                                                                                                                                                        global_config)
            Bases: object
            Base class for decisionengine components such as Sources and Channels
            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
            get_loglevel()
            get_state()
            get_state_name()
            get_state_value()
            set_loglevel_value(log level)
                        Assumes log level is a string corresponding to the supported logging-module levels.
            take_offline(current_data_block)
                        offline and stop this component manager
decisionengine.framework.managers.ComponentManager.create_runner(module_name, class_name,
                                                                                                                                                                               parameters)
            Create instance of dynamically loaded module
decisionengine.framework.managers.SourceManager module
Source Manager
class decisionengine.framework.managers.SourceManager.Source(name, source_dict)
            Bases: object
            Decision Source. Instantiates Source runners according to the provided Source configuration
class decisionengine.framework.managers.SourceManager.SourceManager(name, generation id,
                                                                                                                                                                                        source config,
                                                                                                                                                                                       global_config,
                                                                                                                                                                                        data block queue)
            Bases: \ decision engine. framework. managers. Component \texttt{Manager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.ComponentManager.Compo
            Source Manager: Runs decision cycle for transforms and publishers
            data_block_send(source_name, source_id, data, header)
            run()
```

Provides interface to loadable modules and events for synchronization of execution

# decisionengine.framework.managers.SourceSubscriptionManager module

Source Subscription Manager

#### class

decisionengine.framework.managers.SourceSubscriptionManager.SourceSubscriptionManager Bases: threading.Thread

This implements the communication between Sources and Channels

get\_new\_subscriptions()

run()

Method representing the thread's activity.

You may override this method in a subclass. The standard run() method invokes the callable object passed to the object's constructor as the target argument, if any, with sequential and keyword arguments taken from the args and kwargs arguments, respectively.

send\_data\_product\_to\_subscribed(new\_source\_info)

Bases: object

#### **Module contents**

decisionengine.framework.modules package

**Subpackages** 

decisionengine.framework.modules.tests package

#### **Submodules**

decisionengine.framework.modules.tests.test\_EmptySource module

```
decisionengine.framework.modules.tests.test_EmptySource.test_empty_source_structure()
decisionengine.framework.modules.tests.test_EmptySource.test_missing_data_product_name_not_supported()
```

source\_names)

### decisionengine.framework.modules.tests.test\_Module module

decisionengine.framework.modules.tests.test\_Module.test\_module\_structure()
The module.Module itself is a bit of a skeleton...

#### decisionengine.framework.modules.tests.test Publisher module

decisionengine.framework.modules.tests.test\_Publisher.test\_publisher\_structure()
The module.publisher itself is a bit of a skeleton...

# decisionengine.framework.modules.tests.test\_Source module

decisionengine.framework.modules.tests.test\_Source.test\_source\_structure()
The module.Source itself is a bit of a skeleton...

# decisionengine.framework.modules.tests.test\_Transform module

decisionengine.framework.modules.tests.test\_Transform.test\_transform\_structure()
The module.Transform itself is a bit of a skeleton...

# decisionengine.framework.modules.tests.test\_de\_logger module

```
decisionengine.framework.modules.tests.test_de_logger.log_setup()
decisionengine.framework.modules.tests.test_de_logger.test_by_nonsense_is_err(log_setup)
decisionengine.framework.modules.tests.test_de_logger.test_by_size(log_setup)
decisionengine.framework.modules.tests.test_de_logger.test_by_time(log_setup)
```

# decisionengine.framework.modules.tests.test\_module\_decorators module

```
decisionengine.framework.modules.tests.test_module_decorators.test_multiple_consumes_declarations()
decisionengine.framework.modules.tests.test_module_decorators.test_multiple_produces_declarations()
decisionengine.framework.modules.tests.test_module_decorators.test_supports_config()
decisionengine.framework.modules.tests.test_module_decorators.test_wrong_product_names()
decisionengine.framework.modules.tests.test_module_decorators.test_wrong_product_types()
```

### decisionengine.framework.modules.tests.test\_translate\_product\_name module

```
decisionengine.framework.modules.tests.test_translate_product_name.test_translate_all()
decisionengine.framework.modules.tests.test_translate_product_name.test_translate_illegal_characters()
decisionengine.framework.modules.tests.test_translate_product_name.test_translate_none()
decisionengine.framework.modules.tests.test_translate_product_name.test_translate_simple()
decisionengine.framework.modules.tests.test_translate_product_name.test_translate_with_underscores()
```

#### Module contents

#### **Submodules**

#### decisionengine.framework.modules.EmptySource module

This dummy source takes the name of a source datablock from config file as parameter "data\_product\_name" and produces an empty pandas DataFrame as a datablock with that name

```
class decisionengine.framework.modules.EmptySource.EmptySource(config)
    Bases: decisionengine.framework.modules.Source.Source
    _supported_config = {'data_product_name': (<class 'str'>, '', None)}
    acquire()
decisionengine.framework.modules.Module module
class decisionengine.framework.modules.Module.Module(set_of_parameters)
    Bases: object
    A skeleton of a module
    get_data_block()
    get_parameters()
    set_data_block(data block)
decisionengine.framework.modules.Module.consumes(**kwargs)
decisionengine.framework.modules.Module.produces(**kwargs)
decisionengine.framework.modules.Module.verify_products(producer, data)
decisionengine.framework.modules.Publisher module
class decisionengine.framework.modules.Publisher.Parameter(name, type=None, default=None,
```

class decisionengine.framework.modules.Publisher.Publisher(set\_of\_parameters)

Bases: decisionengine.framework.modules.Module.Module

*comment=None*)

Bases: object

\_consumes = {}

```
publish(data_block=None)
    shutdown()
decisionengine.framework.modules.Publisher.consumes(**kwargs)
decisionengine.framework.modules.Publisher.describe(cls, program_options=<class 'decisio-
                                                      nengine.framework.modules.describe.ModuleProgramOptions'>)
decisionengine.framework.modules.Publisher.supports_config(*args)
decisionengine.framework.modules.Source module
class decisionengine.framework.modules.Source.Parameter(name, type=None, default=None,
                                                          comment=None)
    Bases: object
class decisionengine.framework.modules.Source.Source(set_of_parameters)
    Bases: decisionengine.framework.modules.Module.Module
    _produces = {}
    acquire()
    post_create(global_config)
decisionengine.framework.modules.Source.describe(cls, sample config=None)
decisionengine.framework.modules.Source.produces(**kwargs)
decisionengine.framework.modules.Source.supports_config(*args)
decisionengine.framework.modules.SourceProxy module
Fill in data from another channel data block
class decisionengine.framework.modules.SourceProxy.SourceProxy(config)
    Bases: decisionengine.framework.modules.Source.Source
    _get_data(data block, key)
    _supported_config = {'Dataproducts': (<class 'list'>, None, 'List of data products
    to retrieve.'), 'max_attempts': (<class 'int'>, 10, 'Number of attempts allowed to
    fetch products.'), 'retry_interval': (<class 'int'>, 60, 'Number of seconds to wait
    between retries.'), 'source_channel': (<class 'str'>, None, 'Channel from which to
    retrieve data products.')}
    acquire()
         Overrides Source class method
    post_create(global_config)
```

### decisionengine.framework.modules.Transform module

```
class decisionengine.framework.modules.Transform.Parameter(name, type=None, default=None,
                                                                 comment=None)
     Bases: object
class decisionengine.framework.modules.Transform.Transform(set_of_parameters)
     Bases: decisionengine.framework.modules.Module.Module
     _{consumes} = \{\}
     _produces = {}
     transform()
decisionengine.framework.modules.Transform.consumes(**kwargs)
decisionengine.framework.modules.Transform.describe(cls, program options=<class 'decisio-
                                                         nengine.framework.modules.describe.ModuleProgramOptions'>)
decisionengine.framework.modules.Transform.produces(**kwargs)
decisionengine.framework.modules.Transform.supports_config(*args)
decisionengine.framework.modules.de_logger module
Logger to use in all modules
decisionengine.framework.modules.de_logger._reset_config()
     Reset the logconf.pylogconfig dictionary
decisionengine.framework.modules.de_logger.get_logger()
     get default logger - "decisionengine" :rtype: logging.Logger - rotating file logger
decisionengine.framework.modules.de_logger.set_logging(log level, file rotate by,
                                                             rotation time unit=D', rotation interval=1,
                                                             max_backup_count=6,
                                                             max\_file\_size=2000000000,
                                                             log_file_name='/tmp/decision_engine_logs/decisionengine.log'
```

#### **Parameters**

- log\_level (str) log level
- **file\_rotate\_by** files rotation by size or by time
- rotation\_time\_unit (str) unit of time for file rotation
- rotation\_interval (int) time in rotation\_time\_units between file rotations
- 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 None

#### decisionengine.framework.modules.describe module

```
class decisionengine.framework.modules.describe.ModuleProgramOptions(module_spec, cls)
    Bases: object
    process_args()

class decisionengine.framework.modules.describe.Parameter(name, type=None, default=None, comment=None)
    Bases: object

decisionengine.framework.modules.describe._par_default(par_type, default_value)
decisionengine.framework.modules.describe._par_type(par_type, default_value)
decisionengine.framework.modules.describe.main_wrapper(cls, program_options=<class 'decisionengine.framework.modules.describe.supports_config(*args)

decisionengine.framework.modules.logging_configDict module

Global Logger config dictionary used by all loggers (in their own subkeys)

decisionengine.framework.modules.print_description module

decisionengine.framework.modules.print_description._print_comment(comment)
decisionengine.framework.modules.print_description.print_type(type or value)
```

```
decisionengine.framework.modules.print_description._print_comment(comment)

decisionengine.framework.modules.print_description._print_type(type_or_value)

decisionengine.framework.modules.print_description._print_value(v)

decisionengine.framework.modules.print_description._spec_from_file_name(filename)

decisionengine.framework.modules.print_description.print_consumes(cls)

decisionengine.framework.modules.print_description.print_produces(cls)

decisionengine.framework.modules.print_description.print_supported_config(module_spec, cls)

decisionengine.framework.modules.print_description.spec_if_main(cls)
```

#### decisionengine.framework.modules.translate\_product\_name module

#### **Module contents**

## decisionengine.framework.taskmanager package

#### **Submodules**

#### decisionengine.framework.taskmanager.ProcessingState module

The ProcessingState class can represent any of the following task-manager states:

#### BOOT IDLE ACTIVE STEADY OFFLINE SHUTTINGDOWN SHUTDOWN ERROR

In addition, the class supports 'wait\_until(state)' and 'wait\_while(state)' methods, which, when called from a different process, block until the state has been entered or exited, respectively.

The 'RUNNING\_CONDITIONS' list is a list of states that a thread may have if it is started/starting. The 'STOPPING\_CONDITIONS' list is a list of states that a thread may have if it is stopped/stopping. The 'INAC-TIVE\_CONDITIONS' list is a list of states that a thread may have when it is not active

This object tracks the state of a process.

A number of convience wrappers are provided.

Additionally you may use the .lock attribute for with block to lock the state during specific operations.

#### get()

This function is a minimally locking check to fetch the state.

```
get_state_value()
has_value(state)
inactive()
property lock
probably_running()
set(state)
    This function will lock (and possibly block) to ensure a consistent change to the state value.
    This function can be blocked using the .lock to force state sync between threads if need be.
should_stop()
wait_until(state, timeout=None)
wait_while(state, timeout=None)
class decisionengine.framework.taskmanager.ProcessingState.State(value)
Bases: enum.Enum
An enumeration.
```

ACTIVE = 2 BOOT = 0 ERROR = 7 IDLE = 1

```
SHUTTINGDOWN = 4
     STEADY = 3
decisionengine.framework.taskmanager.TaskManager module
Task Manager
class decisionengine.framework.taskmanager.TaskManager.Channel_dict, channel_name)
     Bases: object
     Decision Channel. Instantiates workers according to channel configuration
class decisionengine.framework.taskmanager.TaskManager.TaskManager(name, generation_id,
                                                                            channel_dict, global_config)
     Bases: decisionengine.framework.managers.ComponentManager.ComponentManager
     Task Manager
     decision_cycle()
          Decision cycle to be run periodically (by trigger)
     do_backup()
          Duplicate current data block and return its copy
              Return type DataBlock
     get_consumes()
     get_produces()
     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.
```

 $OFFLINE = 6 \\
SHUTDOWN = 5$ 

```
Parameters data_block (DataBlock) – data block
     set_to_shutdown()
     start_sources(data_block=None)
          Start sources, each in a separate thread
             Parameters data_block (DataBlock) – data block
     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
class decisionengine.framework.taskmanager.TaskManager.Worker(conf_dict, base_class,
                                                                     channel_name)
     Bases: object
     Provides interface to loadable modules an events to sycronise execution
decisionengine.framework.taskmanager.TaskManager._create_module_instance(config_dict,
                                                                                 base class,
                                                                                 channel_name)
     Create instance of dynamically loaded module
decisionengine.framework.taskmanager.TaskManager._find_only_one_subclass(module, base class)
     Search through module looking for only one subclass of the supplied base_class
decisionengine.framework.taskmanager.TaskManager._make_workers_for(configs, base_class,
                                                                           channel_name)
decisionengine.framework.taskmanager.module_graph module
Ensure no circularities in produces and consumes.
decisionengine.framework.taskmanager.module_graph._consumed_products(*worker_lists)
decisionengine.framework.taskmanager.module_graph._produced_products(*worker_lists)
decisionengine.framework.taskmanager.module_graph.ensure_no_circularities(sources, transforms,
                                                                                  publishers)
     Ensures no circularities among data products.
Module contents
decisionengine.framework.tests package
Submodules
decisionengine.framework.tests.ABTransform module
class decisionengine.framework.tests.ABTransform.ABTransform(module_parameters, *args,
                                                                    **kwargs)
     Bases: decisionengine.framework.modules.Transform.Transform
```

```
_consumes = {'B': None}
     _produces = {'A': None}
decisionengine.framework.tests.BATransform module
class decisionengine.framework.tests.BATransform.BATransform(module_parameters, *args,
                                                                  **kwargs)
     Bases: decisionengine.framework.modules.Transform.Transform
     _consumes = {'A': None}
     _produces = {'B': None}
decisionengine.framework.tests.ErrorOnAcquire module
class decisionengine.framework.tests.ErrorOnAcquire.ErrorOnAcquire(config)
     Bases: decisionengine.framework.modules.Source.Source
     _produces = {'_placeholder': None}
     acquire()
decisionengine.framework.tests.FailingPublisher module
\textbf{class} \ \ \text{decisionengine.framework.tests.} Failing Publisher. \textbf{\textit{FailingPublisher}} (\textit{module\_parameters}, \\
                                                                             *args, **kwargs)
     Bases: decisionengine.framework.modules.Publisher.Publisher
     _consumes = {'bar': None}
     publish(data_block)
decisionengine.framework.tests.FailingSourceNOP module
class decisionengine.framework.tests.FailingSourceNOP.SourceWithMissingProduces(set_of_parameters)
     Bases: decisionengine.framework.modules.Source.Source
decisionengine.framework.tests.FailingSourceProxy module
class decisionengine.framework.tests.FailingSourceProxy.FailingSourceProxy(config)
     Bases: decisionengine.framework.modules.SourceProxy.SourceProxy
         Overrides Source class method
```

### decisionengine.framework.tests.ModuleProgramOptions module

```
class decisionengine.framework.tests.ModuleProgramOptions.AcquireWithConfig(name)
    Bases: object
    test(byte_str, expected_stderr=")
class decisionengine.framework.tests.ModuleProgramOptions.AcquireWithSampleConfig(name)
    Bases: object
    test()
class decisionengine.framework.tests.ModuleProgramOptions.ConfigTemplate(name)
    Bases: object
    test(has comments=False)
class decisionengine.framework.tests.ModuleProgramOptions.Describe(name)
    Bases: object
    test(consumes=None, produces=None)
class decisionengine.framework.tests.ModuleProgramOptions.DescribeAlias(alias, original)
    Bases: object
    test()
class decisionengine.framework.tests.ModuleProgramOptions.Help(name)
    Bases: object
    test(has_sample_config=False)
decisionengine.framework.tests.ModuleProgramOptions._expected_acquire_result(name, con-
                                                                                fig_file=None,
                                                                                multiplier=1,
                                                                                chan-
                                                                                nel name='test1')
decisionengine.framework.tests.ModuleProgramOptions._expected_config_template(name)
decisionengine.framework.tests.ModuleProgramOptions._expected_config_template_with_comments(name)
decisionengine.framework.tests.ModuleProgramOptions._expected_help(name)
decisionengine.framework.tests.ModuleProgramOptions._expected_source_help(name,
                                                                             has_sample_config=False)
decisionengine.framework.tests.ModuleProgramOptions._normalize(string)
decisionengine.framework.tests.ModuleProgramOptions._run_as_main(name, *program_options)
decisionengine.framework.tests.PublisherNOP module
class decisionengine.framework.tests.PublisherNOP.PublisherNOP(module_parameters, *args,
                                                                  **kwargs)
    Bases: decisionengine.framework.modules.Publisher.Publisher
    _consumes = {'bar': <class 'pandas.core.frame.DataFrame'>}
    publish(data_block)
```

#### decisionengine.framework.tests.PublisherWithMissingConsumes module

class decisionengine.framework.tests.PublisherWithMissingConsumes.PublisherWithMissingConsumes(set\_of\_para Bases: decisionengine.framework.modules.Publisher.Publisher
decisionengine.framework.tests.SourceAlias module

#### decisionengine.framework.tests.SourceNOP module

```
class decisionengine.framework.tests.SourceNOP.SourceNOP(config)
    Bases: decisionengine.framework.modules.Source.Source
    _produces = {'foo': <class 'pandas.core.frame.DataFrame'>}
    acquire()
```

# decisionengine.framework.tests.SourceWithSampleConfigNOP module

```
class decisionengine.framework.tests.SourceWithSampleConfigNOP.SourceWithSampleConfigNOP(config)
    Bases: decisionengine.framework.modules.Source.Source
    _produces = {'foo': <class 'pandas.core.frame.DataFrame'>}
    _supported_config = {'channel_name': (<class 'str'>, None, None), 'multiplier': (<class 'int'>, None, None)}
    acquire()
```

#### decisionengine.framework.tests.SupportsConfigPublisher module

```
class decisionengine.framework.tests.SupportsConfigPublisher.SupportsConfig(set_of_parameters)
    Bases: decisionengine.framework.modules.Publisher.Publisher

_supported_config = {'comment': (<class 'str'>, None, 'Single-line comment'),
    'comment_with_nl': (<class 'str'>, None, 'Comment with newline\n'), 'convert_to':
    (<class 'int'>, 3, None), 'default_only': (<class 'float'>, 2.5, None), 'no_type':
    (None, None, None), 'only_type': (<class 'int'>, None, None)}
```

#### decisionengine.framework.tests.TransformNOP module

transform(data\_block)

### decisionengine.framework.tests.TransformWithMissingProducesConsumes module

class decisionengine.framework.tests.TransformWithMissingProducesConsumes.Transf

transform(data\_block)

#### decisionengine.framework.tests.WorkingSourceProxy module

class decisionengine.framework.tests.WorkingSourceProxy.WorkingSourceProxy(config)

 $Bases:\ decisionengine.framework.modules.SourceProxy.SourceProxy$ 

acquire()

Overrides Source class method

#### decisionengine.framework.tests.fixtures module

defaults for pytest

decisionengine.framework.tests.fixtures.**DEServer**(conf\_path=None, conf\_override=None, channel\_conf\_path=None, channel\_conf\_override=None, host='127.0.0.1'.

channel\_conf\_override=None, host='127.0.0.1',
port=None)

A DE Server using a private database

decisionengine.framework.tests.fixtures.PG\_DE\_DB\_WITHOUT\_SCHEMA(request:

\_pytest.fixtures.FixtureRequest)

 $\rightarrow$ 

psycopg2.extensions.connection

Fixture factory for PostgreSQL.

Parameters request - fixture request object

Returns postgresql client

decisionengine.framework.tests.fixtures.**PG\_DE\_DB\_WITH\_SCHEMA**(*PG\_DE\_DB\_WITHOUT\_SCHEMA*) Load our PG schema into the database via this fixture so pytest knows the limitations on parallel usage of this database scope.

decisionengine.framework.tests.fixtures.PG\_PROG(request: \_pytest.fixtures.FixtureRequest,

*tmpdir\_factory:* \_*pytest.tmpdir.TempdirFactory*) → Iterator[pytest\_postgresql.executor.PostgreSQLExecutor]

Process fixture for PostgreSQL.

**Parameters** request – fixture request object

Returns tcp executor

decisionengine.framework.tests.fixtures.SQLALCHEMY\_PG\_WITH\_SCHEMA(PG\_DE\_DB\_WITHOUT\_SCHEMA) Get a blank database from pytest\_postgresql. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

decisionengine.framework.tests.fixtures.**SQLALCHEMY\_TEMPFILE\_SQLITE**(*tmp\_path*)

Setup an SQLite database with the pytest tmp\_path fixture. Then setup the SQLAlchemy style URL with that DB. The SQLAlchemyDS will create the schema as needed.

### decisionengine.framework.tests.test\_client\_errors module

Fixture based DE Server tests of the sample config

Fixture based DE Server for the de-client tests

decisionengine.framework.tests.test\_client\_errors.test\_client\_cannot\_wait\_on\_bad\_state(deserver) Verify wait is for a valid state

#### decisionengine.framework.tests.test client server module

```
decisionengine.framework.tests.test_client_server.test_client_print_product(deserver)

decisionengine.framework.tests.test_client_server.test_client_print_product_columns(deserver)

decisionengine.framework.tests.test_client_server.test_client_print_product_columns_query(deserver)

decisionengine.framework.tests.test_client_server.test_client_print_product_json(deserver)

decisionengine.framework.tests.test_client_server.test_client_print_product_not_real(deserver)

decisionengine.framework.tests.test_client_server.test_client_print_product_not_string(deserver)
```

decisionengine.framework.tests.test\_client\_server.**test\_client\_print\_product\_not\_string**(deserver)

Make sure the public API is protected against bad values

 ${\tt decisionengine.framework.tests.test\_client\_server.test\_client\_print\_product\_query({\it deserver})}$ 

decisionengine.framework.tests.test\_client\_server.test\_client\_print\_product\_types(deserver)

decisionengine.framework.tests.test\_client\_server.test\_client\_print\_product\_vertical(deserver)

decisionengine.framework.tests.test\_client\_server.test\_client\_status\_msg\_to\_stdout(deserver)

Make sure the actual client console call goes to stdout

#### decisionengine.framework.tests.test defaults module

Fixture based DE Server tests of the sample config

decisionengine.framework.tests.test\_defaults.test\_client\_can\_get\_de\_server\_show\_channel\_logger\_level(des Verify unknown channel has NOTSET

decisionengine.framework.tests.test\_defaults.test\_client\_de\_config\_is\_json(deserver) Verify config can be fetched in json format

# decisionengine.framework.tests.test error on acquire module

decisionengine.framework.tests.test\_error\_on\_acquire.test\_source\_only\_channel(deserver)

## decisionengine.framework.tests.test\_module\_program\_options module

```
decisionengine.framework.tests.test_module_program_options.test_acquire_for_sources()
decisionengine.framework.tests.test_module_program_options.test_config_templates()
decisionengine.framework.tests.test_module_program_options.test_descriptions()
decisionengine.framework.tests.test_module_program_options.test_help()
decisionengine.framework.tests.test_module_program_options.test_module_alias()
```

# decisionengine.framework.tests.test\_query\_tool\_server module

```
Fixture based DE Server for the de-query-tool tests

decisionengine.framework.tests.test_query_tool_server.test_query_tool_csv(deserver)

decisionengine.framework.tests.test_query_tool_server.test_query_tool_default(deserver)

decisionengine.framework.tests.test_query_tool_server.test_query_tool_invalid_product(deserver)

decisionengine.framework.tests.test_query_tool_server.test_query_tool_json(deserver)

decisionengine.framework.tests.test_query_tool_server.test_query_tool_since(deserver)
```

#### decisionengine.framework.tests.test reaper module

Fixture based DE Server for the reaper tests

```
decisionengine.framework.tests.test_reaper.test_client_can_get_de_server_reaper_start_delay(deserver) Verify reaper can start with delay
```

decisionengine.framework.tests.test\_reaper.test\_client\_can\_get\_de\_server\_reaper\_status(deserver) Verify reaper status

decisionengine.framework.tests.test\_reaper.test\_client\_can\_get\_de\_server\_reaper\_stop(deserver) Verify reaper can stop

#### decisionengine.framework.tests.test restart channel module

```
decisionengine.framework.tests.test_restart_channel.deserver_mock_data_block(mock_data_block)
decisionengine.framework.tests.test_restart_channel.test_restart_channel(deserver_mock_data_block)
```

# decisionengine.framework.tests.test sample config module

Fixture based DE Server tests of the defaults

```
decisionengine.framework.tests.test_sample_config.test_client_can_double_set_de_server_channel_log_leve Verify set log level to current level isn't an error
```

```
decisionengine.framework.tests.test_sample_config.test_client_can_get_de_server_channel_config(deserver) Verify config has expected items
```

decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_de\_server\_channel\_log\_level(deserver) Verify can fetch log level for a channel

- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_de\_server\_show\_config(deserver) Verify config has expected items
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_de\_server\_show\_logger\_level(deserver) Verify can fetch log level
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_de\_server\_status(deserver) Verify channel enters stable state
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_products(deserver)

  Verify client can get channel products
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_get\_products\_no\_channels(deserver)

  Verify client can get channel products even when none are run
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_kill\_one\_channel(deserver) Verify client can kill a single channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_kill\_one\_channel\_force(deserver) Verify client can kill a single channel with force
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_kill\_one\_channel\_timeout(deserver)

  Verify client can kill a single channel with timeout
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_set\_de\_server\_channel\_log\_level(deserver) Verify set log level for a channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_start\_all\_channel(deserver) Verify client can start all channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_start\_one\_channel(deserver) Verify client can start a single channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_stop\_channels(deserver) Verify client can stop channels
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_stop\_one\_channel(deserver) Verify client can stop a single channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_can\_stop\_server(deserver) Verify de-client can run -stop
- decisionengine.framework.tests.test\_sample\_config.test\_client\_cannot\_double\_start(deserver) Verify client cannot double start channels
- decisionengine.framework.tests.test\_sample\_config.test\_client\_get\_non\_real\_channel(deserver) Verify config for missing channel does what it should
- decisionengine.framework.tests.test\_sample\_config.test\_client\_set\_channel\_log\_fails\_cleanly(deserver) Verify graceful fail on bogus channel
- decisionengine.framework.tests.test\_sample\_config.test\_client\_start\_non\_real\_channel(deserver) Verify start for missing channel does what it should
- decisionengine.framework.tests.test\_sample\_config.test\_client\_stop\_non\_real\_channel(deserver) Verify stop for missing channel does what it should
- decisionengine.framework.tests.test\_sample\_config.test\_client\_wait\_timeout\_works(deserver) Verify channel enters stable state and timeout works too

#### decisionengine.framework.tests.test\_source\_proxy module

Fixture based tests of the SourceProxy module.

```
decisionengine.framework.tests.test_source_proxy.test_stop_failing_source_proxy(deserver_fail) decisionengine.framework.tests.test_source_proxy.test_working_source_proxy(deserver)
```

#### decisionengine.framework.tests.test\_start\_with\_bad\_channels module

```
Fixture based DE Server tests of invalid channel configs
```

```
decisionengine.framework.tests.test_start_with_bad_channels._consumes_not_subset(test_str)
decisionengine.framework.tests.test_start_with_bad_channels._expected_circularity(test_str)
decisionengine.framework.tests.test_start_with_bad_channels._missing_consumes(name)
decisionengine.framework.tests.test_start_with_bad_channels._missing_produces(name)
decisionengine.framework.tests.test_start_with_bad_channels.test_client_can_get_products_no_channels(design)
```

Verify client can get channel products even when none are run

#### decisionengine.framework.tests.test\_start\_with\_no\_channels module

Fixture based DE Server tests of the server without channels, then with them decisionengine.framework.tests.test\_start\_with\_no\_channels.deserver\_mock\_data\_block(mock\_data\_block) decisionengine.framework.tests.test\_start\_with\_no\_channels.test\_start\_from\_nothing(deserver\_mock\_data\_block)

#### Module contents

#### decisionengine.framework.util package

#### **Submodules**

## decisionengine.framework.util.fs module

decisionengine.framework.util.fs.files\_with\_extensions(dir\_path, \*extensions)

Return all files in dir\_path that match the provided extensions.

If no extensions are given, then all files in dir\_path are returned.

Results are sorted by channel name to ensure stable output.

#### decisionengine.framework.util.metrics module

```
class decisionengine.framework.util.metrics.Counter(name, documentation, labelnames=(),
                                                            namespace=", subsystem=", unit=", reg-
                                                            istry=eprometheus_client.registry.CollectorRegistry
                                                            object>, _labelvalues=None)
     Bases: prometheus_client.metrics.Counter
class decisionengine.framework.util.metrics.Gauge(*args, **kwargs)
     Bases: prometheus_client.metrics.Gauge
     Override prometheus client Gauge so that muliproccess_mode 'liveall' is the default as opposed to 'all'
     _DEFAULT_MULTIPROC_MODE = 'liveall'
     __determine_multiprocess_mode_existence(*args, **kwargs)
class decisionengine.framework.util.metrics.Histogram(name, documentation, labelnames=(),
                                                              namespace=", subsystem=", unit=", reg-
                                                              istry=eprometheus_client.registry.CollectorRegistry
                                                              object>, _labelvalues=None, buckets=(0.005,
                                                              0.01, 0.025, 0.05, 0.075, 0.1, 0.25, 0.5, 0.75,
                                                              1.0, 2.5, 5.0, 7.5, 10.0, inf))
     Bases: prometheus_client.metrics.Histogram
class decisionengine.framework.util.metrics.Summary(name, documentation, labelnames=(),
                                                            namespace=", subsystem=", unit=", reg-
                                                            istry=<prometheus client.registry.CollectorRegistry
                                                            object>, labelvalues=None)
     Bases: prometheus_client.metrics.Summary
```

#### decisionengine.framework.util.reaper module

A stand-alone script purges data in database older than specified in configuration. Configuration file has to have this bit added:

```
{
    "dataspace" [{ "retention_interval_in_days"][365,]
    "datasource": { ... }
}
```

Can be used in a cron job.

decisionengine.framework.util.reaper.main()

### decisionengine.framework.util.singleton module

```
class decisionengine.framework.util.singleton.ScopedSingleton
     Bases: decisionengine.framework.util.singleton.Singleton
     Singleton pattern using Metaclass with weak refs
     _instances = <WeakValueDictionary>
class decisionengine.framework.util.singleton.ScopedSingletonABC(name, bases, namespace,
     Bases: abc.ABCMeta, decisionengine.framework.util.singleton.ScopedSingleton
class decisionengine.framework.util.singleton.Singleton
     Bases: type
     Singleton pattern using Metaclass with strong refs
     _instances = {}
class decisionengine.framework.util.singleton.SingletonABC(name, bases, namespace, **kwargs)
     Bases: abc.ABCMeta, decisionengine.framework.util.singleton.Singleton
decisionengine.framework.util.sockets module
decisionengine.framework.util.sockets.get_random_port()
decisionengine.framework.util.subclasses module
decisionengine.framework.util.subclasses._derived_class(cls, base_class)
     Only matches subclasses that are not equal to the base class.
decisionengine.framework.util.subclasses.all_subclasses(module, base_class)
     Return all of a module's subclasses of the given base class.
Module contents
Submodules
decisionengine.framework.about module
PEP-0396 provides instructions for providing module versions While we are at it, add a few other useful bits
decisionengine.framework.version module
Module contents
decisionengine.tests package
```

**Submodules** 

# decisionengine.tests.test\_framework\_package module

Make sure decisionengine.framework is a valid python package decisionengine.tests.test\_framework\_package.test\_can\_import()

#### **Module contents**

## **Module contents**

# 4.2 Indices and tables

- genindex
- modindex
- · search

4.2. Indices and tables 97

# CHAPTER

# **FIVE**

# **INDICES AND TABLES**

- genindex
- modindex
- search

# **PYTHON MODULE INDEX**

```
d
                                                                                                                                                                                                                        decisionengine.framework.dataspace.datasources.tests.test
decisionengine, 97
                                                                                                                                                                                                                        decisionengine.framework.dataspace.dataspace,
decisionengine.framework, 96
decisionengine.framework.about, 96
                                                                                                                                                                                                                        decisionengine.framework.dataspace.maintain,
decisionengine.framework.config, 37
decisionengine.framework.config.ChannelConfigHandler, 64
                                                                                                                                                                                                                        decisionengine.framework.dataspace.tests, 59
                                                                                                                                                                                                                        decisionengine.framework.dataspace.tests.fixtures,
 decisionengine.framework.config.policies, 36
decisionengine.framework.config.tests, 35
\label{lem:decisionengine} decisionengine. framework. config. tests. test\_config. sionengine. framework. dataspace. tests. test\_datablock,
\tt decisionengine.framework.config.tests.test\_de\_decisionengine.framework.dataspace.tests.test\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zincest.est\_datablock\_zinc
decisionengine. framework. config. tests. test\_poldecies; one ngine. framework. dataspace. tests. test\_datasource, tests. tests\_datasource, tests. tests\_datasource, tests_datasource, tests_d
                                                                                                                                                                                                                        decisionengine.framework.dataspace.tests.test_dataspace,
decisionengine.framework.config.ValidConfig,
                                                                                                                                                                                                                        decisionengine.framework.dataspace.tests.test_Reaper,
decisionengine.framework.dataspace, 65
decisionengine.framework.dataspace.datablock,
                                                                                                                                                                                                                        decisionengine.framework.engine, 70
{\tt decisionengine.framework.dataspace.datasource,} {\tt decisionengine.framework.engine.de\_client,70}
                                                                                                                                                                                                                         decisionengine.framework.engine.de_query_tool,
decisionengine.framework.dataspace.datasources
                                                                                                                                                                                                                         decisionengine.framework.engine.DecisionEngine,
decisionengine.framework.dataspace.datasources.null,
                                                                                                                                                                                                                         decisionengine.framework.engine.tests, 66
 decisionengine.framework.dataspace.datasourcesdecisionengine.framework.engine.tests.fixtures,
decisionengine.framework.dataspace.datasources.sulaichemv us: framework.engine.tests.test_client_only,
decisionengine.framework.dataspace.datasourcesdecisionengine.framework_engine.tests.test_query_tool_only
decisionengine.framework.dataspace.datasourcesdecisionengine.framework.engine.tests.test_startup,
decisionengine.framework.dataspace.datasources equal chemy ne.framework.engine.Workers, 69
                                                                                                                                                                                                                       decisionengine.framework.logicengine, 75
decision engine. framework. dataspace. datasources \\ \frac{decision engine. framework. logic engine. Boolean Expression, framework. logic engine. Boolean Expression engine. Boolean Expr
\label{lem:decisionengine} decisionengine. framework. dataspace. datasources \\ decisionengine. framework. logicengine. Fact Lookup, \\ decisionengine. framework. logicengine. \\ decisionengine. \\ decisionen
decisionengine.framework.dataspace.datasourcesdecisionengine.framework.plogicengine.LogicEngine,
                                     48
```

```
decisionengine.framework.logicengine.Rule,74
decisionengine.framework.logicengine.RuleEngindecisionengine.framework.modules.tests.test_module_decorate
decisionengine.framework.logicengine.tests,
                                                                                         decisionengine.framework.modules.tests.test_Publisher,
decisionengine.framework.logicengine.tests.tesdebissilonfermentinenframework.modules.tests.test_Source,
decision engine. framework.logicengine. tests. test \underline{crission dendy involents}, a memork.modules. tests. test \underline{Transform}, and the first test \underline{Transform}, and \underline{Transform},
decisionengine.framework.logicengine.tests.tesdecissistmenogiinoon,framework.modules.tests.test_translate_prod
decisionengine.framework.logicengine.tests.tesdedixpildiocartorifacframework.modules.Transform,
decisionengine.framework.logicengine.tests.tesdecfasciospengine.framework.modules.translate_product_name,
decisionengine.framework.logicengine.tests.tesdecfisilomenceinnorframework.taskmanager,86
                                                                                         decisionengine.framework.taskmanager.module_graph,
decisionengine.framework.logicengine.tests.test_pandas6fact,
                                                                                         decisionengine.framework.taskmanager.ProcessingState,
decisionengine.framework.logicengine.tests.test_rule_with_negated_fact,
                                                                                         decisionengine.framework.taskmanager.TaskManager,
decisionengine.framework.logicengine.tests.test_simple5configuration,
                                                                                         decisionengine.framework.tests,94
decisionengine.framework.managers, 78
                                                                                         decisionengine.framework.tests.ABTransform.
decisionengine.framework.managers.ChannelManager,
                                                                                          decisionengine.framework.tests.BATransform,
decisionengine.framework.managers.ComponentManager,
                                                                                          decisionengine.framework.tests.ErrorOnAcquire,
decisionengine.framework.managers.SourceManager,
                                                                                          decisionengine.framework.tests.FailingPublisher,
decisionengine.framework.managers.SourceSubscriptionManager,
                                                                                          decisionengine.framework.tests.FailingSourceNOP,
decisionengine.framework.modules, 84
decisionengine.framework.modules.de_logger,
                                                                                          decisionengine.framework.tests.FailingSourceProxy,
decisionengine.framework.modules.describe, 83
                                                                                         decisionengine.framework.tests.fixtures, 90
decisionengine.framework.modules.EmptySource, decisionengine.framework.tests.ModuleProgramOptions,
decisionengine.framework.modules.logging_confideditionengine.framework.tests.PublisherNOP,
decisionengine.framework.modules.Module, 80
                                                                                          decisionengine.framework.tests.PublisherWithMissingConsume
decisionengine.framework.modules.print_description,
                                                                                          decisionengine.framework.tests.SourceAlias,
decisionengine.framework.modules.Publisher,
                                                                                          decisionengine.framework.tests.SourceNOP, 89
decisionengine.framework.modules.Source, 81
                                                                                          decisionengine.framework.tests.SourceWithSampleConfigNOP,
decisionengine.framework.modules.SourceProxy,
                                                                                         decisionengine.framework.tests.SupportsConfigPublisher,
decisionengine.framework.modules.tests, 80
decisionengine.framework.modules.tests.test_deddoissionengine.framework.tests.test_client_errors,
decisionengine.framework.modules.tests.test_EmphacyTssinnovengine.framework.tests.test_client_server,
decisionengine.framework.modules.tests.test_ModMexdesionengine.framework.tests.test_defaults,
```

102 Python Module Index

```
91
decisionengine.framework.tests.test_error_on_acquire,
decisionengine.framework.tests.test_module_program_options,
decisionengine.framework.tests.test_query_tool_server,
decisionengine.framework.tests.test_reaper,
decisionengine.framework.tests.test_restart_channel,
decisionengine.framework.tests.test_sample_config,
decisionengine.framework.tests.test_source_proxy,
decisionengine.framework.tests.test_start_with_bad_channels,
decisionengine.framework.tests.test_start_with_no_channels,
decisionengine.framework.tests.TransformNOP,
decisionengine.framework.tests.TransformWithMissingProducesConsumes,
decisionengine.framework.tests.WorkingSourceProxy,
decisionengine.framework.util, 96
decisionengine.framework.util.fs, 94
decisionengine.framework.util.metrics, 95
decisionengine.framework.util.reaper, 95
decisionengine.framework.util.singleton, 96
decisionengine.framework.util.sockets, 96
decisionengine.framework.util.subclasses, 96
decisionengine.framework.version, 96
decisionengine.tests, 97
decisionengine.tests.test_framework_package,
```

Python Module Index 103

104 Python Module Index

## **INDEX**

DEFAULT_MULTIPROC_MODE
tribute), 95 determine_multiprocess_mode_existence()
determine_multiprocess_mode_existence()
(decisionengine.framework.util.metrics.Gauge method), 95 —insert() (decisionengine.framework.dataspace.datablock.DataBrockdecisionengine.framework.modules.Transform.Transform method), 59 —query() (decisionengine.framework.dataspace.datasources.posymes.decisionengine.framework.tests.ABTransform.ABTransform attribute), 80  —query() (decisionengine.framework.dataspace.datasources.posymes.decisionengine.framework.tests.ABTransform.ABTransform attribute), 86
method), 95 insert() (decisionengine.framework.dataspace.datablock.Datablock.decisionengine.framework.modules.Transform.Transform method), 59 query() (decisionengine.framework.dataspace.datasources.posyges (decisionengine.framework.tests.ABTransform.ABTransform method), 52 query(), 80 query() (decisionengine.framework.dataspace.datasources.posyges (decisionengine.framework.tests.ABTransform.ABTransform method), 53
method), 59 query() (decisionengine.framework.dataspace.datasourecs.posygresql.psignes.gine.framework.tests.ABTransform.ABTransform  method), 52  attribute), 82 query() (decisionengine.framework.dataspace.datasourecs.posygresql.psignes.gine.framework.tests.ABTransform.ABTransform  method), 52
method), 59 query() (decisionengine.framework.dataspace.datasourecs.posygresql.psignes.gine.framework.tests.ABTransform.ABTransform  method), 52  attribute), 82 query() (decisionengine.framework.dataspace.datasourecs.posygresql.psignes.gine.framework.tests.ABTransform.ABTransform  method), 52
mathod) 52 attribute), 86
update() (decisionengine.framework.dataspace.datablock.DataBlock decisionengine.framework.tests.BATransform.BATransform method), 59 attribute), 87
_abc_impl (decisionengine.framework.config.ValidConfig.ValidConfig. (decisionengine.framework.tests.FailingPublisher.FailingPubli
_abc_impl (decisionengine.framework.dataspace.datablock_freader)
attribute) 60 attribute), 88
_abc_impl (decisionengine.framework.dataspace.datablock.Metadata (decisionengine.framework.tests.TransformNOP.TransformNOP attribute), 60 attribute), 89
_abc_impl (decisionengine.framework.dataspace.datasource.DataSource not_subset() (in module decisio-attribute), 61 nengine.framework.tests.test_start_with_bad_channels),
_abc_impl (decisionengine.framework.dataspace.datasources.null.NullDataSource attribute), 50create_de_server() (in module decisio-
_abc_impl (decisionengine.framework.dataspace.datasources.postgresquesquesquesquesquesquesquesquesquesqu
_abc_impl (decisionengine.framework.dataspace.datasourees:squatchenty_as:dataframework.logicengine.LogicEngine attribute), 44  nengine.framework.logicengine.LogicEngine
_abc_impl (decisionengine.framework.dataspace.datasources.sqlalchemyods:datasource_api.SQLAlchemyDS attribute), 37create_module_instance() (in module decisio-
_channel_config_dir() (in module decisio- nengine.framework.config.tests.test_config),  nengine.framework.taskmanager.TaskManager),  86
dataframe_to_column_names() (decisio-
_channel_preamble() (in module decisio- nengine.framework.engine.DecisionEngine
nengine.framework.engine.DecisionEngine), method), 66
dataframe_to_csv() (decisio-
_check_keys() (in module decisio- nengine.framework.config.ChannelConfigHandler), nengine.framework.engine.DecisionEngine.DecisionEngine method), 66
ataframe_to_json() (decisio-
_check_override() (in module decisio- nengine.framework.engine.DecisionEngine
nengine.framework.engine.tests.test_startup), method), 6/ 66dataframe_to_table() (decisio-
_config_from_file() (in module decisio- nengine.framework.engine.DecisionEngine.DecisionEngine

method), 67		channel()		(decisio-
_dataframe_to_vertical_tables()	(decisio-		onfig.ChannelCo	onfigH and ler. Channel ConfigH
nengine.framework.engine.Decisionl				
method), 67		de_logger() (in	module	decisio-
_delete() (decisionengine.framework.datasp	ace.datasources.post	gr <b>exalgPoetfrærsæl</b> work.co	onfig.ChannelCo	onfigHandler),
method), 52	1	35	1 1	7
_derived_class() (in module	decisiomake_			decisio-
<pre>nengine.framework.util.subclasses), _dispatch()</pre>	(decisio-	nengine.framework.m 76	anagers.Cnanne	umanager),
nengine.framework.engine.Decisionl	`		n module	decisio-
method), 67	Site the Decision	nengine.framework.ta		
_expected_acquire_result() (in modul	e decisio-	86	~	
nengine.framework.tests.ModuleProg		ng_consumes() (in	n module	decisio-
88		nengine.framework.te	sts.test_start_wi	th_bad_channels),
_expected_circularity() (in module		94		
nengine.framework.tests.test_start_w	ith_bad_chan <b>neils</b> )și			decisio-
94		nengine.framework.te	sts.test_start_wi	th_bad_channels),
_expected_config_template() (in modul		94	1 1	1
nengine.framework.tests.ModuleProg	gramOptions), norma	*	module	decisio-
88 _expected_config_template_with_comme	n+c()	nengine.framework.te. 88	sis.MoauteProgi	ramOpiions),
(in module		lefault() (in	module	decisio-
nengine.framework.tests.ModuleProg	•	nengine.framework.m		
88	_par_t		module	decisio-
_expected_help() (in module	decisio-	nengine.framework.m		
nengine.framework.tests.ModuleProg	gramOptions) <mark>.</mark> print		module	decisio-
88		nengine.framework.m	odules.print_des	scription),
_expected_source_help() (in module		83		
nengine.framework.tests.ModuleProg	g <i>ramOptions</i> ) <u>.</u> print		module	decisio-
88	7	nengine.framework.m	odules.print_des	scription),
_find_only_one_subclass() (in module nengine.framework.taskmanager.Task		83	module	decisio-
86	kimanager), _print	:_value() (in nengine.framework.m		
_get_data()	(decisio-	83	ounies.prini_ue.	scripitoti);
nengine.framework.modules.SourceI	•		in module	decisio-
method), 81	, — <b>.</b> ,	nengine.framework.ta		ule_graph),
_get_de_conf_manager() (in module	decisio-	86		
nengine.framework.engine.Decisionl	Engine), _produ	ices (decisionengine.fra	mework.modules	s.Source.Source
68		attribute), 81		
_get_global_config() (in module		ices (decisionengine.fra	mework.modules	s.Transform.Transform
nengine.framework.engine.Decisionl		attribute), 82		DT f A DT f
_global_config_file() (in module	_produ decisio-	attribute), 87	mework.tests.AB	BTransform.ABTransform
_global_config_file() (in module nengine.framework.config.tests.test_			mework tests RA	Transform.BATransform
33	_prode	attribute), 87	nework.tests.br	iransjorm. <b>D</b> riransjorm
_insert() (decisionengine.framework.datasp	ace.datasourc <b>esmodst</b>		mework.tests.Er	rorOnAcquire.ErrorOnAcq
method), 52	<b>-1</b> 1	attribute), 87		1
_insert_returning_result()	(decisioprodu	ices (decisionengine.fra	mework.tests.So	urceNOP.SourceNOP
nengine.framework.dataspace.datasa				
method), 52			mework.tests.So	urce With Sample Config NO.
_instances(decisionengine.framework.util.st			. –	4
attribute), 96	_		mework.tests.Tra	ansformNOP.TransformNO
_instances(decisionengine.framework.util.su		attribute), 89		(dagisia
attribute), 96	_reape	er_loop()		(decisio-

```
nengine.framework.dataspace.maintain.Reaper
                                                                                                                                                                              nengine.framework.tests.SourceWithSampleConfigNOP.SourceWi
                        method), 64
                                                                                                                                                                              attribute), 89
_remove() (decisionengine.framework.dataspace.datasourcesumpopreedl.Powkiesql
                                                                                                                                                                              nengine.framework.tests.SupportsConfigPublisher.SupportsConfig
                        method), 52
_reset_config()
                                                                (in
                                                                                        module
                                                                                                                           decisio-
                                                                                                                                                                              attribute), 89
                        nengine.framework.modules.de logger),
                                                                                                                                                     _update() (decisionengine.framework.dataspace.datasources.postgresql.F
                        82
                                                                                                                                                                              method), 52
                                                                                                                                                     _update_channel_states()
_run_as_main()
                                                              (in
                                                                                       module
                                                                                                                           decisio-
                        nengine.framework.tests.ModuleProgramOptions),
                                                                                                                                                                              nengine.framework.engine.Workers.Workers
                                                                                                                                                                              method), 69
                        88
_sa_class_manager
                                                                                                                         (decisio- _update_returning_result()
                                                                                                                                                                                                                                                                               (decisio-
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.dhenglinmfraDatesproddataspace.datasources.postgresql.Postgresql
                                                                                                                                                                              method), 52
                        attribute), 41
_sa_class_manager
                                                                                                                         (decisio-
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.db_schema.Header
                        attribute), 41
                                                                                                                                                     ABTransform
                                                                                                                                                                                                                (class
                                                                                                                                                                                                                                                                                 decisio-
                                                                                                                         (decisio-
_sa_class_manager
                                                                                                                                                                               nengine.framework.tests.ABTransform), 86
                        nengine. framework. datas pace. datas our ces. sqlalch \c the limit of the limit 
                        attribute), 42
                                                                                                                                                                              method), 69
_sa_class_manager
                                                                                                                         (decisio-
                                                                                                                                                     acquire() (decisionengine.framework.modules.EmptySource.EmptySource
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.dbnesqhanaschema
                        attribute), 43
                                                                                                                                                     acquire() (decisionengine.framework.modules.Source.Source
_sa_class_manager
                                                                                                                         (decisio-
                                                                                                                                                                               method), 81
                        nengine.framework.dataspace.datasources.sqlalchami. ประชาบารสาราชาธิการสาราชาธิการสาราชาธิการสาราชาธิการสาราชา
                        attribute), 43
                                                                                                                                                                              method), 81
_sa_registry
                                                                                                                         (decisio-
                                                                                                                                                   acquire() (decisionengine.framework.tests.ErrorOnAcquire.ErrorOnAcqui
                        nengine. framework. dataspace. datasources. sqlalchemy\_ds. db_{ne} gbernas Base
                        attribute), 41
                                                                                                                                                     acquire() (decisionengine.framework.tests.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxy.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.FailingSourceProxY.Failin
_select() (decisionengine.framework.dataspace.datasources.postgresql
                        method), 52
                                                                                                                                                     acquire() (decisionengine.framework.tests.SourceNOP.SourceNOP
_select_dictresult()
                                                                                                                         (decisio-
                                                                                                                                                                              method), 89
                        nengine. framework. dataspace. datas our ces. postgress of PPFECTS (alecision engine. framework. tests. Source With Sample ConfigNOP) and the configNOP of th
                        method), 52
                                                                                                                                                                              method), 89
_select_getresult()
                                                                                                                         (decisio-
                                                                                                                                                     acquire() (decisionengine.framework.tests.WorkingSourceProxy.Working
                        nengine.framework.dataspace.datasources.postgresql.Postgresqhod), 90
                        method), 52
                                                                                                                                                     AcquireWithConfig
                                                                                                                                                                                                                             (class
                                                                                                                                                                                                                                                            in
_select_tuple()
                                                                                                                         (decisio-
                                                                                                                                                                               nengine.framework.tests.ModuleProgramOptions),
                        nengine. framework. data space. data sources. postgresql. Postgresql\\
                                                                                                                                                     AcquireWithSampleConfig
                                                                                                                                                                                                                                          (class
                                                                                                                                                                                                                                                                                 decisio-
\_setitem() (decisionengine.framework.dataspace.datablock.ar{D}ataBlockfigine.framework.tests.ModuleProgramOptions),
                        method), 59
_spec_from_file_name()
                                                                               (in
                                                                                               module
                                                                                                                          decisio-
                                                                                                                                                     \verb"ACTIVE" (decision engine. framework. taskmanager. Processing State. State
                        nengine.framework.modules.print_description),
                                                                                                                                                                              attribute), 84
                        83
                                                                                                                                                     add_engine_pidguard()
                                                                                                                                                                                                                                    (in
                                                                                                                                                                                                                                                     module
                                                                                                                                                                                                                                                                                 decisio-
_start_de_server()
                                                                       (in
                                                                                           module
                                                                                                                           decisio-
                                                                                                                                                                              nengine.framework.dataspace.datasources.sqlalchemy_ds.utils),
                        nengine.framework.engine.DecisionEngine),
                                                                                                                                                                               43
                        68
                                                                                                                                                     all_subclasses()
                                                                                                                                                                                                                         (in
                                                                                                                                                                                                                                               module
                                                                                                                                                                                                                                                                                 decisio-
_supported_config
                                                                                                                         (decisio-
                                                                                                                                                                               nengine.framework.util.subclasses), 96
                        nengine.framework.modules.EmptySource.EmptySource
                        attribute), 80
_supported_config
                                                                                                                         (decisio-
                                                                                                                                                                                                 (class
                                                                                                                                                                                                                                                                                 decisio-
                                                                                                                                                      Base
                                                                                                                                                                                                                                              in
                        nengine.framework.modules.SourceProxy.SourceProxy
                                                                                                                                                                              nengine.framework.dataspace.datasources.sqlalchemy_ds.db_sch
                        attribute), 81
_supported_config
                                                                                                                         (decisio-
```

BATrans		(class	in	decisio-		61			
		nework.tests.I	BATransforn		config(		(in	module	decisio-
block_u				(decisio-			.framework	config.tests.test_d	e_std),
		nework.engin	e.DecisionE	Ingine.Deci	_				
	method), 67				config(		(in	module	decisio-
block_w				(decisio-			.framework	dataspace.tests.te	st_Reaper),
		nework.engin	e.DecisionE	ingine.Deci.	_		/ 1		7
D 1	method), 67			7	ConfigT			ass in	decisio-
Rootean	Expression	`	in Maina Baala	decisio-	:)	nengine. 88	.jrameworк	t.tests.ModuleProg	ramOptions),
	nengine.jrai 73	nework.logice	епдіпе.Бооів	eanExpressi			ionanaina:	framowork datasno	nea datasourea DataSourea
DOOT (da	, 0	framework.ta	skmanagar l	Drocassina				татемотк.аанаѕрс	ace.datasource.DataSource
воот (ие	attribute), 8		skmunuger.1	rocessings				framework datasno	ace.datasources.null.NullDa
	airibuie), o	T			Connect	method)		тате поткланазре	ice.aaiasoarces.naii.ivaiiDa
С					connect			framework datasna	ace.datasources.postgresql.I
Channel	(a	lass	in	decisio-	Comicce	method)		тате поткланазре	ice.aaiasourees.posigresqi.1
Chamier	nanaina fra	iuss nework mana	in aars Chann	uecisio- alManagar)	connect			framework.dataspa	ace.datasources.sqlalchemy_
	75	nework.mana	gers.Channe	envianager)	,	method)	, 37		,
Channel		lass	in	decisio-	connect			framework.dataspa	ace.datasources.sqlalchemy_
	,	nework.taskm				method)	, 44	_	
	85				console	_script	s_main()	(in module	decisio-
channel	_config_di	r() (in	module	decisio-				engine.de_client)	, 70
	_	nework.config	g.policies), 3	86	console	_script	s_main()	(in module	decisio-
Channel	ConfigHand		_	decisio-		nengine	.framework	engine.de_query_	tool),
	nengine.fran	nework.config	g.ChannelCo	onfigHandle		70			
	35				consume		_	r.framework.logice	ngine.LogicEngine.LogicEn
Channel		(class	in	decisio-		method)			
	nengine.fran	nework.mana	gers.Channe	elManager)	, consume	s()	(in	module	decisio-
	75							modules.Module)	
check_f	or_new_dat	a_products	()	(decisio-	consume	s()	(in	module	decisio-
			gers.Channe	elManager.0	ChannelM	a <b>nagg</b> yne.	.jrameworк	.modules.Publishe	r),
(1	method), 75				acomanmo	81	(in	module	decisio
Clean (d		e.framework.e	engine.Decis	sionEngine.	StopState	nanaina	(in	moaute c.modules.Transfor	decisio-
-1	attribute), 6			1		82	.jramework	mounies.11unsjo1	m),
clone_m	odel()		nodule nasa datasa	decisio-	Counter		(class	in	decisio-
	nengine.jrai 43	nework.datas <sub>i</sub>	pace.aaiaso	urces.sqiaic	nemy <u>·</u> us:	uuus), nenoine		.util.metrics), 95	accisio
close()	(decisionene	rina framawor	k datasnaca	datasource	oreate			module	decisio-
(1036()	method), 61	ine.jramewor	к.ишизрисе	·uuiusource	Duiusoi <del>a</del>	nengine.	.framework	.engine.de_client)	, 70
close()		gine.framewor	k dataspace	datasource	SCHARA TARL	_	•		decisio-
C105C()	method), 50		к.аагазрасс	.uarasource	25.110111.114011	nengine.	.framework	.engine.de_query_	tool),
close()		gine.framewor	k.dataspace	.datasource	es.postgres	_		· -1 · -	
()	method), 52				create_			module	decisio-
close()		gine.framewor	k.dataspace	.datasource	es.sqlalche	m <u>engine</u> 77	dravare <u>r</u> k	apn:SIQEATChemyD)	<b>g</b> entManager),
close()		gine.framewor	k datasnaca	datasourca	create.	tables(	Alchemy	$2\Omega_{\rm v}$	(decisio-
(1036()	method), 44		к.ишизрисе	·uuiusource	is.sqiaien <del>e</del>	nengine.	.framework	dataspace.dataso	
close()		gine.framewor	k datasnace	datasnace	DataSpac	_		1	
C103C()	method), 63		к.аанаграсс	.иинизрисс.	create_	tables(	Ć		(decisio-
Compone	ntManager	(class	in	decisio-				dataspace.dataso	urces.null.NullDataSource
Joinpoile.	_	nework.mana			er),	method)	*	•	
	77				create_	tables(	$\mathcal{C}$		(decisio-
compres	s()	(in me	odule	decisio-		nengine	.framework	dataspace.dataso	urces.postgresql.Postgresql
•		nework.datas <sub>i</sub>	pace.datable	ock),		method)	, 52		

```
create_tables()
                                                                                                (decisio-
                                                                                                                                          method), 85
                   nengine.framework.dataspace.datasources.sqlalchdooi_sliothemspimee_api.SQLAlchemyDS
                                                                                                                                module, 97
create_tables()
                                                                                                (decisio- DecisionEngine
                                                                                                                                                                          (class
                                                                                                                                                                                                     in
                                                                                                                                                                                                                        decisio-
                   method), 44
                                                                                                                                          66
                                                                                                (decisio- decisionengine.framework
create time
                   nengine.framework.dataspace.datasources.sqlalchemy mbchhl eschoma.Header
                   attribute), 42
                                                                                                                      decisionengine.framework.about
{\tt creator} ({\it decision engine. framework. dataspace. datasources. sqlattor dataspace. datasources. sqlattor dataspace. datasources. sqlattor dataspace. datasources. sqlattor dataspace. dataspace
                   attribute), 42
                                                                                                                      decisionengine.framework.config
                                                                                                                                module, 37
D
                                                                                                                      decisionengine.framework.config.ChannelConfigHandler
                                                                                                                                module, 35
data_block_put()
                                                                                                (decisio-
                   nengine.framework.managers.ComponentManagedecinjonenginegeramework.config.policies
                                                                                                                                module, 36
                   method), 77
                                                                                                                      decisionengine.framework.config.tests
data_block_send()
                                                                                                (decisio-
                   nengine.framework.managers.SourceManager.SourceManager.35
                                                                                                                      decisionengine.framework.config.tests.test_config
                   method), 77
                                                                                                                                module, 33
DataBlock
                                           (class
                                                                          in
                                                                                                  decisio-
                                                                                                                      {\tt decisionengine.framework.config.tests.test\_de\_std}
                   nengine.framework.dataspace.datablock),
                                                                                                                                module, 34
                                                                                                                      decisionengine.framework.config.tests.test_policies
Dataproduct
                                              (class
                                                                           in
                                                                                                  decisio-
                   nengine.framework.dataspace.datasources.sqlalchemy mandal eschema),
                                                                                                                      decisionengine.framework.config.ValidConfig
                                                                                                                                module, 35
dataproduct_table
                                                                                                (decisio-
                   \textit{nengine.} framework. data space. data source. Data Sode \texttt{c} \texttt{isionengine.} framework. datas \texttt{pace}
                                                                                                                                module, 65
                   attribute), 61
                                                                                                                      decisionengine.framework.dataspace.datablock
                                                                                                 decisio-
DataSource
                                            (class
                                                                           in
                                                                                                                                module, 59
                   nengine.framework.dataspace.datasource),
                                                                                                                      decisionengine.framework.dataspace.datasource
                   61
                                                                                                                                module, 61
datasource()
                                              (in
                                                                   module
                                                                                                 decisio-
                   \textit{nengine.} \textit{framework.} \textit{dataspace.} \textit{datasources.} \textit{tests.} \textit{fideses}, \textbf{jonengine.} \textit{framework.} \textit{dataspace.} \textit{datasources.} 
                                                                                                                                module, 55
                                                                                                                      decisionengine.framework.dataspace.datasources.null
datasource()
                                              (in
                                                                   module
                                                                                                  decisio-
                                                                                                                                module, 50
                   nengine.framework.dataspace.tests.fixtures),
                                                                                                                      decisionengine.framework.dataspace.datasources.postgresql
                    55
                                                                                                                                module, 52
                                                                                                 decisio-
DataSpace
                                           (class
                                                                          in
                                                                                                                      decisionengine.framework.dataspace.datasources.sqlalchemy_
                   nengine.framework.dataspace.dataspace),
                                                                                                                                module, 44
                   63
                                                                                                                      decisionengine.framework.dataspace.datasources.sqlalchemy_
dataspace()
                                                                   module
                                                                                                  decisio-
                                                                                                                                module, 37
                   nengine.framework.dataspace.tests.fixtures),
                                                                                                                      decisionengine.framework.dataspace.datasources.sqlalchemy_
                                                                                                                                module, 41
DataSpaceConfigurationError, 64
                                                                                                                      decisionengine.framework.dataspace.datasources.sqlalchemy_
DataSpaceConnectionError, 64
                                                                                                                                module, 43
DataSpaceError, 64
                                                                                                                      decisionengine.framework.dataspace.datasources.tests
DataSpaceExistsError, 64
datestamp (decisionengine, framework. dataspace. datasources. softall half of the first of the f
                                                                                                                      decisionengine.framework.dataspace.datasources.tests.fixtu
                   attribute), 43
                                                                                                                                module, 47
decision_cycle()
                                                                                                (decisio-
                   nengine.framework.managers.ChannelManager.Channelmanager.channengine.framework.dataspace.datasources.tests.test
                                                                                                                                module, 48
                   method), 75
                                                                                                                      decisionengine.framework.dataspace.datasources.tests.test_
decision_cycle()
                                                                                                (decisio-
```

nengine.framework.taskmanager.TaskManager.TaskManager.50

```
decisionengine.framework.dataspace.dataspace decisionengine.framework.logicengine.tests.test_cascaded_1
    module, 63
                                                    module, 71
decisionengine.framework.dataspace.maintain
                                                decisionengine.framework.logicengine.tests.test_constructi
    module, 64
                                                    module, 71
decisionengine.framework.dataspace.tests
                                                decisionengine.framework.logicengine.tests.test_duplicate_
   module, 59
                                                    module, 71
decisionengine.framework.dataspace.tests.fixturbesisionengine.framework.logicengine.tests.test_facts
    module, 55
                                                    module, 71
decisionengine.framework.dataspace.tests.test_dlactisbilomorkngine.framework.logicengine.tests.test_fail_on_en
    module, 57
                                                    module, 72
decisionengine.framework.dataspace.tests.test_datasplomekngzilnieb.framework.logicengine.tests.test_pandas_fac
    module, 57
                                                    module, 72
decisionengine.framework.dataspace.tests.test_ddatasponenegine.framework.logicengine.tests.test_rule_with_
    module, 57
                                                    module, 72
decisionengine.framework.dataspace.tests.test_dbatiaspancengine.framework.logicengine.tests.test_simple_cor
    module, 58
                                                    module, 72
decisionengine.framework.dataspace.tests.test_Reaperionengine.framework.managers
    module, 56
                                                   module, 78
decisionengine.framework.engine
                                                decisionengine.framework.managers.ChannelManager
    module, 70
                                                    module, 75
decisionengine.framework.engine.de_client
                                                decisionengine.framework.managers.ComponentManager
                                                    module, 77
decisionengine.framework.engine.de_query_tool decisionengine.framework.managers.SourceManager
    module, 70
                                                    module, 77
decisionengine.framework.engine.DecisionEnginedecisionengine.framework.managers.SourceSubscriptionManage
    module, 66
                                                    module, 78
decisionengine.framework.engine.tests
                                                decisionengine.framework.modules
    module, 66
                                                    module, 84
decisionengine.framework.engine.tests.fixturesdecisionengine.framework.modules.de_logger
    module, 65
                                                    module, 82
\tt decisionengine.framework.engine.tests.test\_cliebert\_sindry engine.framework.modules.describe
    module, 66
                                                    module, 83
decisionengine.framework.engine.tests.test_quedrycitsiodnemgine.framework.modules.EmptySource
    module, 66
                                                   module, 80
decisionengine.framework.engine.tests.test_stadecopsionengine.framework.modules.logging_configDict
   module, 66
                                                   module, 83
decisionengine.framework.engine.Workers
                                                decisionengine.framework.modules.Module
    module, 69
                                                    module, 80
decisionengine.framework.logicengine
                                                decisionengine.framework.modules.print_description
    module, 75
                                                    module, 83
decisionengine.framework.logicengine.BooleanExpereissionmengine.framework.modules.Publisher
    module, 73
                                                    module, 80
decisionengine.framework.logicengine.FactLookupecisionengine.framework.modules.Source
    module, 73
                                                    module, 81
decisionengine.framework.logicengine.LogicEngidhecisionengine.framework.modules.SourceProxy
    module, 74
                                                    module, 81
decisionengine.framework.logicengine.Rule
                                                decisionengine.framework.modules.tests
    module, 74
                                                    module, 80
decisionengine.framework.logicengine.RuleEngindecisionengine.framework.modules.tests.test_de_logger
    module, 75
                                                    module, 79
decisionengine.framework.logicengine.tests
                                                decisionengine.framework.modules.tests.test_EmptySource
    module, 73
                                                    module, 78
decisionengine.framework.logicengine.tests.tesdediosilonfemginionframework.modules.tests.test_Module
    module, 71
                                                    module, 79
```

```
decisionengine.framework.modules.tests.test_modbadiesidemoernginoersframework.tests.test_client_server
    module, 79
                                                    module, 91
decisionengine.framework.modules.tests.test_Publidishionnengine.framework.tests.test_defaults
    module, 79
                                                    module, 91
decisionengine.framework.modules.tests.test_Sombraciesionengine.framework.tests.test_error_on_acquire
                                                    module, 91
   module, 79
decisionengine.framework.modules.tests.test_Trakensifsoirunengine.framework.tests.test_module_program_options
    module, 79
                                                    module, 92
decisionengine.framework.modules.tests.test_trakensilsaiten_enrepibuectf_ramework.tests.test_query_tool_server
    module, 80
                                                    module, 92
decisionengine.framework.modules.Transform
                                                decisionengine.framework.tests.test_reaper
                                                    module, 92
    module, 82
decisionengine.framework.modules.translate_prodexcitsinammengine.framework.tests.test_restart_channel
    module, 83
                                                    module, 92
decisionengine.framework.taskmanager
                                                decisionengine.framework.tests.test_sample_config
    module, 86
                                                    module, 92
decisionengine.framework.taskmanager.module_graphisionengine.framework.tests.test_source_proxy
                                                    module, 94
decisionengine.framework.taskmanager.ProcessindyStinstinonengine.framework.tests.test_start_with_bad_channel
    module, 84
                                                    module, 94
decisionengine.framework.taskmanager.TaskManagakecisionengine.framework.tests.test_start_with_no_channels
                                                    module, 94
decisionengine.framework.tests
                                                decisionengine.framework.tests.TransformNOP
    module, 94
                                                    module, 89
decisionengine.framework.tests.ABTransform
                                                decisionengine.framework.tests.TransformWithMissingProduce
    module, 86
                                                    module, 90
decisionengine.framework.tests.BATransform
                                                decisionengine.framework.tests.WorkingSourceProxy
    module, 87
                                                    module, 90
decisionengine.framework.tests.ErrorOnAcquire decisionengine.framework.util
    module, 87
                                                    module, 96
decisionengine.framework.tests.FailingPublishedecisionengine.framework.util.fs
    module, 87
                                                    module, 94
decisionengine.framework.tests.FailingSourceNOMPecisionengine.framework.util.metrics
    module, 87
                                                    module, 95
decisionengine.framework.tests.FailingSourceProbexyisionengine.framework.util.reaper
   module, 87
                                                    module, 95
decisionengine.framework.tests.fixtures
                                                decisionengine.framework.util.singleton
    module, 90
                                                    module, 96
decisionengine.framework.tests.ModuleProgramOpdeiconsionengine.framework.util.sockets
    module, 88
                                                    module, 96
decisionengine.framework.tests.PublisherNOP
                                                decisionengine.framework.util.subclasses
    module, 88
                                                    module, 96
decisionengine.framework.tests.PublisherWithMidesinspCommessumiesse.framework.version
                                                    module, 96
   module, 89
decisionengine.framework.tests.SourceAlias
                                                decisionengine.tests
    module, 89
                                                    module, 97
decisionengine.framework.tests.SourceNOP
                                                decisionengine.tests.test_framework_package
    module, 89
                                                    module, 97
decisionengine.framework.tests.SourceWithSamplateComfrigeNSP()
                                                                    (in
                                                                            module
                                                                                         decisio-
    module, 89
                                                         nengine.framework.dataspace.datablock),
\tt decisionengine.framework.tests.SupportsConfigPublishe {\bf f}1
    module, 89
                                                default_data_lifetime
decisionengine.framework.tests.test_client_errors
                                                        nengine.framework.dataspace.datablock.Header
    module, 91
                                                         attribute), 60
```

```
delete() (decisionengine, framework.dataspace.dataspace.DataSpacmengine, framework.dataspace.datasource.DataSource
                               method), 63
                                                                                                                                                                                                                         method), 61
delete_data_older_than()
                                                                                                                                                       (decisio- duplicate_datablock()
                               nengine.framework.dataspace.datasource.DataSource
                                                                                                                                                                                                                         nengine.framework.dataspace.datasources.null.NullDataSource
                              method), 61
                                                                                                                                                                                                                         method), 50
delete_data_older_than()
                                                                                                                                                       (decisio- duplicate_datablock()
                                                                                                                                                                                                                                                                                                                                                  (decisio-
                              nengine.framework.dataspace.datasources.null.NullDataSouwaegine.framework.dataspace.datasources.postgresql.Postgresql
                              method), 50
                                                                                                                                                                                                                         method), 52
delete_data_older_than()
                                                                                                                                                       (decisio- duplicate_datablock()
                                                                                                                                                                                                                                                                                                                                                  (decisio-
                              nengine.framework.dataspace.datasources.postgresql.Postgresqgline.framework.dataspace.datasources.sqlalchemy_ds.dataso
                              method), 52
                                                                                                                                                                                                                         method), 38
                                                                                                                                                       (decisio- duplicate_datablock()
delete_data_older_than()
                                                                                                                                                                                                                                                                                                                                                  (decisio-
                              nengine.framework.dataspace.datasources.sqlalchemy_ds.dataspiwacframewofk.dlatuspuyDeSdatasources.sqlalchemy_ds.SQLAl
                              method), 37
                                                                                                                                                                                                                         method), 44
delete_data_older_than()
                                                                                                                                                       (decisio- duplicate_datablock()
                                                                                                                                                                                                                                                                                                                                                  (decisio-
                               nengine.framework.dataspace.datasources.sqlalchemy_ds.SQdn\sirleefranD\swork.dataspace.dataspace.DataSpace
                              method), 44
                                                                                                                                                                                                                         method), 63
Describe
                                                                 (class
                                                                                                                                                          decisio-
                               nengine.framework.tests.ModuleProgramOptions)
                                                                                                                                                                                          EmptySource
                                                                                                                                                                                                                                                                                                                                                    decisio-
                                                                                                                                                                                                                                                                   (class
                                                                                                                                                          decisio-
describe()
                                                                    (in
                                                                                                       module
                                                                                                                                                                                                                         nengine.framework.modules.EmptySource),
                              nengine.framework.modules.Publisher),
                               81
                                                                                                                                                                                          ensure_no_circularities() (in module decisio-
describe()
                                                                                                       module
                                                                                                                                                          decisio-
                                                                    (in
                                                                                                                                                                                                                         nengine.framework.taskmanager.module_graph),
                               nengine.framework.modules.Source), 81
describe()
                                                                   (in
                                                                                                       module
                                                                                                                                                          decisio-
                                                                                                                                                                                          {\tt ERROR}\,(decision engine. framework. task manager. Processing State. State
                               nengine.framework.modules.Transform),
                                                                                                                                                                                                                         attribute), 84
                               82
                                                                                                                                                                                          ErrorOnAcquire
                                                                                                                                                                                                                                                                            (class
                                                                                                                                                                                                                                                                                                                      in
                                                                                                                                                                                                                                                                                                                                                    decisio-
DescribeAlias
                                                                              (class
                                                                                                                                                          decisio-
                                                                                                                          in
                                                                                                                                                                                                                          nengine.framework.tests.ErrorOnAcquire),
                               nengine.framework.tests.ModuleProgramOptions),
                                                                                                                                                                                          \verb|evaluate()| (decision engine. framework. logic engine. Boolean Expression. Between the property of the pro
DEServer()
                                                                    (in
                                                                                                       module
                                                                                                                                                          decisio-
                                                                                                                                                                                                                         method), 73
                               nengine.framework.engine.tests.fixtures),
                                                                                                                                                                                          evaluate() (decisionengine.framework.logicengine.LogicEngine.LogicEn
                                                                                                                                                                                                                         method), 74
DEServer()
                                                                                                       module
                                                                                                                                                          decisio-
                                                                    (in
                                                                                                                                                                                          evaluate() (decisionengine.framework.logicengine.Rule.Rule
                              nengine.framework.tests.fixtures), 90
                                                                                                                                                                                                                         method), 74
deserver_mock_data_block() (in module decisio-
                                                                                                                                                                                          evaluate_facts()
                                                                                                                                                                                                                                                                                                                                                   (decisio-
                               nengine.framework.tests.test_restart_channel),
                                                                                                                                                                                                                         nengine.framework.logicengine.LogicEngine.LogicEngine
                               92
deserver_mock_data_block() (in module decisio-
                                                                                                                                                                                          execute() (decisionengine.framework.logicengine.RuleEngine.RuleEngine
                              nengine.framework.tests.test start with no channels),
                                                                                                                                                                                                                         method), 75
                              94
                                                                                                                                                                                          execute_command_from_args() (in module decisio-
                                                                                                                                                       (decisio-
do_backup()
                                                                                                                                                                                                                         nengine.framework.engine.de_client), 70
                              nengine.framework.managers.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelManager.ChannelMa
                              method), 75
                                                                                                                                                                                                                          nengine.framework.engine.de_query_tool), 70
do_backup()
                                                                                                                                                       (decisio-
                                                                                                                                                                                          expiration_time
                                                                                                                                                                                                                                                                                                                                                  (decisio-
                              nengine. framework. task manager. Task Manager. Task Manager, task Man
                                                                                                                                                                                                                          attribute), 42
{\tt dump()}\ ({\it decisionengine.framework.config.ValidConfig.ValidConfig}\ . ValidConfig. Valid
duplicate()
                                                                                                                                                       (decisio-
                                                                                                                                                                                          FactLookup
                                                                                                                                                                                                                                                                                                                                                    decisio-
                                                                                                                                                                                                                                                                 (class
                                                                                                                                                                                                                                                                                                                 in
                              nengine.framework.dataspace.datablock.DataBlock
                                                                                                                                                                                                                          nengine.framework.logicengine.FactLookup),
                              method), 59
duplicate_datablock()
                                                                                                                                                       (decisio-
```

FailingPublisher (class nengine.framework.tes		<pre>method), 50 get_datablock()</pre>	(decisio-
87	us.1 anng1 nonsner),		(accisio lataspace.datasources.postgresql.Postgresql
FailingSourceProxy (cl	lass in decisio-	method), 53	auaspace.uaiasources.posigresqi.1 osigresqi
	sts.FailingSourceProxy),		(decisio-
nengine.jramework.ies 87	us.FaitingSourcer roxy),		
0,	<i>(</i> : 1.1 1 · · ·		lataspace.datasources.sqlalchemy_ds.dataso
<pre>files_with_extensions()</pre>		method), 38	(1
nengine.framework.uti		<pre>get_datablock()</pre>	(decisio-
<pre>function_name_from_call()</pre>		<u> </u>	lataspace.datasources.sqlalchemy_ds.SQLA
	gicengine.BooleanExpress		
73		<pre>get_datablock()</pre>	(decisio-
G			lataspace.dataspace.DataSpace
_		method), 63	
Gauge (class in decisionengine	?.framework.util.metrics),	get_dataproduct()	(decisio-
95		nengine.framework.d	lataspace.datasource.DataSource
<pre>generate_insert_query()</pre>	(in module decisio-	method), 61	
nengine.framework.da	itaspace.datasources.postg	gr <b>@qt</b> ),dataproduct()	(decisio-
54		nengine.framework.d	lataspace.datasources.null.NullDataSource
generation_id	(decisio-	method), 50	
		ch <b>ga</b> ty_ <b>da:tap_sodenata:D</b> ataprodu	act (decisio-
attribute), 41	1		lataspace.datasources.postgresql.Postgresql
generation_id	(decisio-	method), 53	
		ch <b>ga</b> ty_ <b>dat.Hpsoderat.H</b> eader	(decisio-
attribute), 42	respectiveness in costs quen		lataspace.datasources.sqlalchemy_ds.dataso
generation_id	(decisio-	method), 38	<i>q</i>
		ch <b>ga</b> ty_ <b>da:thpsohwat</b> .Metadata	(decisio-
attribute), 42		nengine.framework.d	lataspace.datasources.sqlalchemy_ds.SQLA
generation_time	(decisio-	method), 45	(1)
	taspace.datasources.sqlal	ch <b>go</b> ty_db:tdp:soldenata.Wetadata	(decisio-
attribute), 42			lataspace.dataspace.DataSpace
<pre>get() (decisionengine.framework)</pre>	rk.dataspace.datablock.De		
method), 59		<pre>get_dataproducts()</pre>	(decisio-
<pre>get() (decisionengine.framewor method), 84</pre>	rk.taskmanager.Processin	gState.Proc <b>ersingiSec</b> freamework.d method), 59	lataspace.datablock.DataBlock
<pre>get_channels()</pre>	(decisio-	<pre>get_dataproducts()</pre>	(decisio-
nengine.framework.co. method), 35	nfig.ChannelConfigHandl	er.Channel <b>Congillafrdine</b> work.d method), 62	lataspace.datasource.DataSource
<pre>get_connection()</pre>	(decisio-	<pre>get_dataproducts()</pre>	(decisio-
			lataspace.datasources.null.NullDataSource
method), 53		method), 50	
<pre>get_consumes()</pre>	(decisio-	<pre>get_dataproducts()</pre>	(decisio-
nengine.framework.en method), 69	gine.Workers.Worker	nengine.framework.d method), 53	lataspace.datasources.postgresql.Postgresql
<pre>get_consumes()</pre>	(decisio-	<pre>get_dataproducts()</pre>	(decisio-
	skmanager.TaskManager.T	TaskManageriengine.framework.d method), 38	lataspace.datasources.sqlalchemy_ds.dataso
<pre>get_data_block()</pre>	(decisio-	<pre>get_dataproducts()</pre>	(decisio-
_	odules.Module.Module		lataspace.datasources.sqlalchemy_ds.SQLA
method), 80	James.Moame.Moame	method), 45	uiuspace.aaiusourees.sqiaienemy_us.5QD/1
get_datablock()	(decisio	<pre>get_dataproducts()</pre>	(decisio-
_			(aecisio- lataspace.dataspace.DataSpace
method), 61	taspace.datasource.DataS	method), 63	
<pre>get_datablock()</pre>	`	<pre>get_header()</pre>	(decisio-
nengine framework da	itasnace datasources null	NullDataSomemoine framework d	latasnace datablock DataRlock

method), 59			method), 51				
<pre>get_header()</pre>	(decisio-	get_me				(decisio-	
nengine.framework.dataspace.dat	asource.DataS	ource	nengine.frame	work.data	space.datas	ources.postgresql.	Postgresql
method), 62			method), 53				
<pre>get_header()</pre>	(decisio-					(decisio-	
nengine.framework.dataspace.dat	asources.null.1	VullDataS	So <b>uwere</b> gine.frame	work.data	ispace.datas	ources.sqlalchemy	_ds.dataso
method), 51			method), 39				
<pre>get_header()</pre>	(decisio-					(decisio-	
nengine.framework.dataspace.dat	asources.postg	resql.Posi		work.data	space.datas	ources.sqlalchemy	_ds.SQLAl
method), 53			method), 46				
<pre>get_header()</pre>			tadata()			(decisio-	
nengine.framework.dataspace.dat	asources.sqlal	chemy_ds.		xSQ1kAllatı	<b>espusiDeS</b> datas <sub>I</sub>	pace.DataSpace	
method), 38			method), 63				
<pre>get_header()</pre>			w_subscription		~	(decisio-	~
nengine.framework.dataspace.dat	asources.sqlal	chemy_ds.		work.man	agers.Source	eSubscriptionMan	ager.Source
method), 45	(1		method), 78			/ 1 · · ·	
<pre>get_header()</pre>			rameters()	1		(decisio-	
nengine.framework.dataspace.dat	aspace.DataSp	ace	nengine.frame	work.moa	ules.Module	.Module	
method), 63	(decisio	aa+ mm	method), 80			(decisio	
get_last_generation_id()	(decisio-	-		nuark ana	in a Wankana	(decisio-	
nengine.framework.dataspace.dat method), 62	asource.Daias	ource	nengine.frame method), 69	work.eng	ne.workers.	worker	
<pre>get_last_generation_id()</pre>	(decisio-					(decisio-	
nengine.framework.dataspace.dat method), 51	asources.null.1	NullDataS	So <b>uwere</b> gine.frame method), 85	work.task	manager.Tas	kManager.TaskMa	ınager
<pre>get_last_generation_id()</pre>	(decisio-	get rai	ndom_port()	(in	module	decisio-	
nengine.framework.dataspace.dat							
method), 53	<b>T 3</b>	get_scl			, ,	(decisio-	
<pre>get_last_generation_id()</pre>	(decisio-	J		work.date	space.datas	ource.DataSource	
nengine.framework.dataspace.dat		chemy_ds.					
method), 39	-	get_scl	_			(decisio-	
<pre>get_last_generation_id()</pre>	(decisio-		nengine.frame	work.data	ispace.datas	ources.null.NullDo	ıtaSource
nengine.framework.dataspace.dat	asources.sqlal	chemy_ds.	.S <b>@ĿtArbod</b> h)e,r <b>ő</b> yDS	S			
method), 45	_	get_scl				(decisio-	
<pre>get_last_generation_id()</pre>	(decisio-		nengine.frame	work.data	space.datas	ources.postgresql.	Postgresql
nengine.framework.dataspace.dat	aspace.DataSp	ace	method), 53				
method), 63		get_scl	hema()			(decisio-	
<pre>get_logger()</pre>	(decisio-		nengine.frame	work.data	ispace.datas	ources.sqlalchemy	_ds.dataso
nengine.framework.engine.Decision	onEngine.Deci	sionEngir	ne method), 39				
method), 67		get_scl	hema()			(decisio-	
<pre>get_logger() (in module</pre>	decisio-		nengine.frame	work.data	ispace.datas	ources.sqlalchemy	_ds.SQLAl
nengine.framework.modules.de_lo	ogger),		method), 46				
82		get_st				(decisio-	
<pre>get_loglevel()</pre>	(decisio-		0 0	work.man	agers.Comp	onentManager.Co	mponentMo
nengine.framework.managers.Com	nponentManag						
method), 77		get_st	ate_name()			(decisio-	
<pre>get_metadata()</pre>	(decisio-	_	nengine.frame	work.eng	ine.Workers.	Worker	
nengine.framework.dataspace.dat	ablock.DataBl		method), 69				
method), 59		get_st	ate_name()	7	~	(decisio-	• -
<pre>get_metadata()</pre>	(decisio-			work.man	agers.Comp	onentManager.Co	mponentMo
nengine.framework.dataspace.dat	asource.DataS		<i>method</i> ), 77			(1:-:	
method), 62	(decisio-	-	ate_value()			(decisio- onentManager Cor	
Der Meradarati	1/10/1510-		nongino tramo	WILLY MAY	HUPES L OMN	$\alpha n \rho n n \nu I (n n n o \rho r I \cap n$	annanant Wia

nengine.framework.dataspace.datasources.null.NullDataSoumethod), 77

```
get_state_value()
                                                                                                                           (decisio- Header
                                                                                                                                                                                                         (class
                                                                                                                                                                                                                                                                                      decisio-
                        nengine.framework.taskmanager.ProcessingState.ProcessingState.framework.dataspace.datablock),
                        method), 84
get_taskmanager()
                                                                                                                           (decisio- Header
                                                                                                                                                                                                         (class
                                                                                                                                                                                                                                                                                      decisio-
                        nengine.framework.dataspace.datablock.DataBlock
                                                                                                                                                                                 nengine.framework.dataspace.datasources.sqlalchemy_ds.db_sch
                        method), 59
get_taskmanager()
                                                                                                                           (decisio- header table
                                                                                                                                                                                                                                                                                   (decisio-
                        nengine. framework. data space. data source. Data Source\\
                                                                                                                                                                                 nengine.framework.dataspace.datasource.DataSource
                        method), 62
                                                                                                                                                                                  attribute), 62
get_taskmanager()
                                                                                                                                                                                                    (class
                                                                                                                           (decisio- Help
                        nengine. framework. dataspace. datasources. null. Null Data Source gine. framework. tests. Module Program Options),\\
                        method), 51
get_taskmanager()
                                                                                                                           (decisio- Histogram
                                                                                                                                                                                                               (class
                                                                                                                                                                                                                                                                                      decisio-
                                                                                                                                                                                                                                                       in
                        nengine.framework.dataspace.datasources.postgresql.Postgresqdine.framework.util.metrics), 95
                        method), 54
                                                                                                                           (decisio-
get_taskmanager()
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.dbtasourceframesQlrk.ltdraspacs.datasources.sqlalchemy_ds.db_sc
                        method), 39
                                                                                                                                                                                 attribute), 41
get_taskmanager()
                                                                                                                           (decisio- \verb| id| (decisionengine.framework.dataspace.datasources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlalchemy\_ds.db\_sclassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassources.sqlassourc
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.SQH-Ajchemy_DS
                        method), 46
                                                                                                                                                       id (decisionengine.framework.dataspace.datasources.sqlalchemy ds.db sc
get_taskmanager()
                                                                                                                           (decisio-
                                                                                                                                                                                  attribute), 42
                        nengine. framework. data space. data space. Data Space (decision engine. framework. task manager. Processing State. State) and the space of the sp
                        method), 64
                                                                                                                                                                                 attribute), 84
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                       inactive() (decisionengine.framework.taskmanager.ProcessingState.Proc
                        nengine.framework.dataspace.datasource.DataSource
                                                                                                                                                                                 method), 84
                        method), 62
                                                                                                                                                        insert() (decisionengine.framework.dataspace.datasource.DataSource
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                                                 method), 62
                        nengine.framework.dataspace.datasources.null.NullPataSourGecisionengine.framework.dataspace.datasources.null.NullData
                        method), 51
                                                                                                                                                                                 method), 51
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                       insert() (decisionengine.framework.dataspace.datasources.postgresql.Po
                        nengine.framework.dataspace.datasources.postgresql.Postgrasqhod), 54
                        method), 54
                                                                                                                                                        insert() (decisionengine.framework.dataspace.datasources.sqlalchemy_a
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                                                  method), 40
                        nengine.framework.dataspace.datasources.sqlalchemy_deligionengineQframework.dataspace.datasources.sqlalchemy_d
                        method), 39
                                                                                                                                                                                  method), 46
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                       \verb"insert(")" (decision engine. framework. data space. data space. Data Space")
                        nengine.framework.dataspace.datasources.sqlalchemy_ds.SQleAlshemy_DS
                        method), 46
                                                                                                                                                        InvalidMetadataError, 60
get_taskmanagers()
                                                                                                                           (decisio-
                                                                                                                                                       is_expired()
                                                                                                                                                                                                                                                                                   (decisio-
                        nengine.framework.dataspace.dataspace.DataSpace
                                                                                                                                                                                 nengine. framework. data space. datablock. Data Block\\
                        method), 64
                                                                                                                                                                                  method), 60
global_config_dir()
                                                                           (in
                                                                                               module
                                                                                                                              decisio-
                                                                                                                                                       is\_valid() (decisionengine.framework.dataspace.datablock.Header
                         nengine.framework.config.policies), 36
                                                                                                                                                                                  method), 60
global_config_file()
                                                                             (in
                                                                                                module
                                                                                                                             decisio-
                                                                                                                                                       K
                         nengine.framework.config.policies), 36
                                                                                                                                                        key (decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_s
Η
                                                                                                                                                                                 attribute), 41
handle_sighup()
                                                                                                                           (decisio-
                                                                                                                                                       key (decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_s
                        nengine.framework.engine.DecisionEngine.DecisionEngine attribute), 42
                        method), 67
                                                                                                                                                        key (decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_s
has_value()
                                                                                                                           (decisio-
                                                                                                                                                                                 attribute), 42
                        nengine. framework. datas pace. datablock. Data Block in the property of the
```

method), 60

method), 84

load() (in module decisionengine framework.config.tests.test_config),	L					73				
Metadata   Cass   in   decisio-   nengine-framework.config. Channel ConfigHandler. Channe	load()	(in	module	decisio-	Metadat		`			
load_all_channels()   decisionengine_framework.config.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.ChannelConfigHandler.channelConfigHan			config.tests.test_c	onfig),			amework.dai	taspace.datab	lock),	
method), 35  load_channel ()	load_al	1 channels()		(decisio-	Metadat	a	(class			
load_channel()		nengine.framework.	config.ChannelCo	onfigHandle	r.Channel	<b>Censins</b> 42	amework.dai	taspace.datas	ources.sqlalchemy_d	ls.db_s
method), 35  load_sample_data_into_datasource()	load_ch	* *		(decisio-	metadat	a (decision	engine.frame	ework.dataspa	ice.datasources.sqla	lchemy
load_sample_data_into_datasource()			config.ChannelCo	onfigHandle	r.Channel metadat	.a_tab1e			`	
mengine.framework.dataspace.tests.fixtures), 56  lock (decisionengine,framework.taskmanager.ProcessingState.ProcestIftENHING.64 property), 84  log_setup() (in module decisionengine.framework.modules.tests.test_de_logger), 79  logic_engine_with_fact() (in module decisionengine.framework.logicengine.tests.test_fail_on_error), 72  logicErgine (class in decisionengine.framework.logicengine.LogicEngine), 74  LogicError, 73  MIN_RETENTION_INTERVAL_DAYS (decisionengine,framework.dataspace.maintain.Reaper) attribute), 64 missed_update_count (decisionengine.framework.dataspace.datasources.sqlalchemy_ds.db_s) attribute), 42 mock_data_block() (in module decisionengine.framework.dataspace.datasources.tests.fixtures), 48 module  decisionengine.framework.dataspace.datasources.tests.fixtures), 48 module  decisionengine.framework.dataspace.datasources.tests.fixtures), 48 module  decisionengine.framework.about, 96 decisionengine.framework.config.37 decisionengine.framework.config.73 decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests.test_config, 32 decisionengine.framework.config.tests.test_config, 32 decisionengine.framework.config.tests.test_config, 32	load_sa	· · · · · · · · · · · · · · · · · · ·	atasource()					taspace.datas	ource.DataSource	
nengine-framework.dataspace.maintain.Reaper		(in mo	odule	decisio-					<i>(</i> <b>1</b>	
lock (decisionengine framework.taskmanager.ProcessingState.Processtificitude). 64			dataspace.tests.fix	xtures),	MIN_RET				`	
log_setup() (in module decisio- nengine.framework.modules.tests.test_de_logger), 79 missed_update_count (decisio- nengine.framework.dataspace.datasources.sqlalchemy_ds.db_s nengine.framework.logicengine.tests.test_fail_on_error), 72 attribute), 42 mock_data_block() (in module decisio- nengine.framework.logicengine.LogicEngine), 74 main() (in module decisio- nengine.framework.engine.de_client), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 main() (in module decisio- nengine.framework.dataspace.datasources.sqlalchemy_ds.db_s nengine.framework.dataspace.datasources.tests.fixtures), 48 module  decisionengine, 97 decisionengine.framework.about, 96 decisionengine.framework.config, 37 decisionengine.framework.config.ChannelConfigHandler 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config, 32  decisionengine.framework.config.tests.test_config,	lock (de	cisionengine.framewo	ork.taskmanager.I	ProcessingS	tate.Proce	. જાદામાં જે ક્યારે છે.	64	P	_	
nengine.framework.modules.tests.test_de_logger), 79 missed_update_count (decisio- nengine.with_fact() (in module decisio- nengine.framework.logicengine.tests.test_fail_on_error), attribute), 42 72 mock_data_block() (in module decisio- nengine.framework.logicengine.LogicEngine), 74 main() (in module decisio- nengine.framework.engine.de_client), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 main() (in module decisio- nengine.framework.engine.de_client), 70 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config, 32	1		1 1	1	IIIN_SEC			taspace maint	`	
missed_update_count (decisio- logic_engine_with_fact() (in module decisio- nengine.framework.logicengine.tests.test_fail_on_error), attribute), 42  72 mock_data_block() (in module decisio- nengine.framework.logicengine.LogicEngine), 74 module  LogicError, 73 main() (in module decisio- nengine.framework.engine.de_client), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 main() (in module decisio- nengine.framework.dataspace.datasources.sqlalchemy_ds.db_s nengine.framework.dataspace.datasources.tests.fixtures), 48 module decisionengine.framework, 96 decisionengine.framework.config, 37 decisionengine.framework.config.policies, 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config,	1og_seτ	_			`			ensperce.menni	am.reaper	
logic_engine_with_fact() (in module decisionengine.framework.logicengine.tests.test_fail_on_error), attribute), 42  To attribute), 42  mock_data_block() (in module decisionengine.framework.logicengine.LogicEngine), attribute), 42  mock_data_block() (in module decisionengine.framework.dataspace.datasources.tests.fixtures), 48  module  LogicError, 73  Main() (in module decisionengine.framework.engine.de_client), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config, 33  decisionengine.framework.config.tests.test_config, 33  decisionengine.framework.config.tests.test_config, 33			moauies.iesis.iesi <u>.</u>	_ae_togger	, missed_				(decisio-	
nengine.framework.logicengine.tests.test_fail_on_error), attribute), 42 72 mock_data_block() (in module decisio- mock_data_block() (in module decisio- nengine.framework.logicengine.LogicEngine), 74 module  LogicError, 73 decisionengine.framework.about, 96 main() (in module decisio- nengine.framework.engine.de_client), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 main() (in module decisio- nengine.framework.engine.de_query_tool), 70 decisionengine.framework.config.ChannelConfigHandler 35 decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config,	logic 6	• •	) (in module	decisio-		_		taspace.datas	ources.sqlalchemy_d	ls.db_s
TogicEngine (class in decisionengine.framework.logicengine.LogicEngine), 74  LogicError, 73  M  main() (in module decisionengine.framework.engine.de_client), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.config.policies, 36  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config, 33  decisionengine.framework.config.tests.test_config, 33  decisionengine.framework.config.tests.test_config, 33	TOGIC_E				error).				•	
mengine.framework.logicengine.LogicEngine), 74  LogicError, 73  M  main() (in module decisionengine.framework.engine.de_client), 70  main() (in module decisionengine.framework.engine.de_client), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.config.policies, 36  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config.			o green ginemesisi.	.0510111011.	mock_da					
module  LogicError, 73  M  main() (in module decisionengine.framework.engine.de_client), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.config.policies, 36  decisionengine.framework.config.policies, 36  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config.	LogicEr	igine (class	in	decisio-			amework.da	taspace.datas	ources.tests.fixtures),	,
LogicError, 73  M main() (in module decisionengine.framework.about, 96 main() (in module decisionengine.framework.engine.de_client), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config, 37			logicengine.Logic	Engine),	module	48				
decisionengine.framework, 96 decisionengine.framework.about, 96 decisionengine.framework.about, 96 decisionengine.framework.about, 96 decisionengine.framework.config, 37 decisionengine.framework.config.ChannelConfigHandler 35 decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests,35 decisionengine.framework.config.tests.test_config,	LogicEr	* *				_				
main() (in module decisionengine.framework.engine.de_client), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.config.policies, 36 decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests, 35		,				_				
main() (in module decisionengine.framework.engine.de_client), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.engine.de_query_tool), 70 main() (in module decisionengine.framework.config.policies, 36 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config.	M					_				
main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.engine.de_query_tool), 36  main() (in module decisionengine.framework.engine.DecisionEngine), decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config.	main()	(in	module	decisio-		_		_		
main() (in module decisionengine.framework.engine.de_query_tool), 70  main() (in module decisionengine.framework.config.policies, 36  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config.		nengine.framework.	engine.de_client)	, 70	dec	_	ine.iramew	ork.coniig.	.ChannelConfigHa	ndler
main() (in module decisionengine.framework.engine.DecisionEngine),  mengine.framework.engine.DecisionEngine),  36  decisionengine.framework.config.tests, 35  decisionengine.framework.config.tests.test_config.	main()	`			doc		no framou	ork config	nolicies	
main() (in module decisionengine.framework.config.tests, 35 decisionengine.framework.config.tests.test_config,			engine.de_query_	_tool),	uec		ille. II alliew	ork.coming.	.porrcres,	
nengine.framework.engine.DecisionEngine),  decisionengine.framework.config.tests.test_config,			1 1	,	dec		ne.framew	ork.confia.	tests, 35	
	main()	nengine.framework.				_		_		ig,
main() (in module decisio- decisionengine.framework.config.tests.test_de_std,	main()	(in		decisio-	dec	_	ine.framew	ork.config.	.tests.test_de_s	td,
nengine. Jramework. util. reaper), 95			_	1	dec		ne.framew	ork.config.	.tests.test_poli	cies.
main_wrapper() (in module decisio- nengine.framework.modules.describe), 83	main_wr	"				_		J		,
make_db() (in module decisio- decisionengine.framework.config.ValidConfig,	make dh				dec	isionengi	ine.framew	ork.config.	.ValidConfig,	
nengine.framework.logicengine.tests.test_facts).	marc_an	•								
decisionengine.framework.dataspace, 65		~ .		, , , , , , , , , , , , , ,		_		_		
mark_demented()	mark_de	emented()		(decisio-	dec	_	ine.framew	ork.dataspa	ace.datablock,	
nengine.framework.dataspace.dataspace.DataSpace 59		nengine.framework.	dataspace.datasp	ace.DataSp	ace		·			
method), 64 decisionengine.framework.dataspace.datasource,		method), 64			aec		ine.framew	ork.dataspa	ace.datasource,	
mark_expired() (decisio- mark_expired() (decis	mark_ex	-		*	doc		no framow	ork dataens	aca datasourcas	
пендіне. Ітаніе work. айнамоск. Динамоск			dataspace.datable	ock.DataBle	ock uec		ille. II alliew	ork.uacaspa	ice.uacasources,	
method), 60		* *		(1 ::	dec		ne.framew	ork.datasna	ace.datasources	nu]].
mark_expired() (decisio-	mark_ex	=	dataanaan data	`		_	amen	caope		,
method), 64  method), 64  decisionengine.framework.dataspace.datasources.postg	_	method), 64			dec dec	isionengi	ine.framew	ork.dataspa	ace.datasources.	postg
maybe_fail_on_error() (in module decisio-nengine.framework.logicengine.BooleanExpression), decisionengine.framework.dataspace.datasources.sqlal	maybe_f		`		on). dec		ine.framew	ork.dataspa	ace.datasources.	sqlal

```
44
                                              decisionengine.framework.logicengine.FactLookup,
decisionengine.framework.dataspace.datasources.sqTalchemy_ds.datasource_api,
                                              decisionengine.framework.logicengine.LogicEngine,
decisionengine.framework.dataspace.datasources.sqTalchemy_ds.db_schema,
                                              decisionengine.framework.logicengine.Rule,
decisionengine.framework.dataspace.datasources.sqTalchemy_ds.utils,
                                              decisionengine.framework.logicengine.RuleEngine,
decisionengine.framework.dataspace.datasources.tes/ts,
                                              decisionengine.framework.logicengine.tests,
decisionengine.framework.dataspace.datasources.tes/ts.fixtures,
                                              decisionengine.framework.logicengine.tests.test_bool_f
decisionengine.framework.dataspace.datasources.testds.test_datasource_api,
                                              decisionengine.framework.logicengine.tests.test_cascad
decisionengine.framework.dataspace.datasources.tests.test_postgresql,
                                              decisionengine.framework.logicengine.tests.test_constr
decisionengine.framework.dataspace.dataspace,
                                              decisionengine.framework.logicengine.tests.test_duplic
decisionengine.framework.dataspace.maintain,
                                              decisionengine.framework.logicengine.tests.test_facts,
decisionengine.framework.dataspace.tests,
                                              decisionengine.framework.logicengine.tests.test_fail_c
decisionengine.framework.dataspace.tests.fixtures,72
                                              decisionengine.framework.logicengine.tests.test_pandas
decisionengine.framework.dataspace.tests.test_datablock,
                                              decisionengine.framework.logicengine.tests.test_rule_w
decisionengine.framework.dataspace.tests.test_datablock_zlib,
                                              decisionengine.framework.logicengine.tests.test_simple
decisionengine.framework.dataspace.tests.test_dataSource,
                                              decisionengine.framework.managers, 78
decisionengine.framework.dataspace.tests.test_dkatiaspacengine.framework.managers.ChannelManager,
decisionengine.framework.dataspace.tests.test_Reaperionengine.framework.managers.ComponentManager,
decisionengine.framework.engine, 70
                                              decisionengine.framework.managers.SourceManager,
decisionengine.framework.engine.de_client,
                                              decisionengine.framework.managers.SourceSubscriptionMa
decisionengine.framework.engine.de_query_tool,
                                              decisionengine.framework.modules, 84
decisionengine.framework.engine.DecisionEnginedecisionengine.framework.modules.de_logger,
decisionengine.framework.engine.tests, 66
                                              decisionengine.framework.modules.describe,
decisionengine.framework.engine.tests.fixtures,
                                              decisionengine.framework.modules.EmptySource,
decisionengine.framework.engine.tests.test_client_@nly,
                                              decisionengine.framework.modules.logging_configDict,
decisionengine.framework.engine.tests.test_query_tool_only,
                                              decisionengine.framework.modules.Module,
decisionengine.framework.engine.tests.test_startup@0
                                              decisionengine.framework.modules.print_description,
decisionengine.framework.engine.Workers,
                                              decisionengine.framework.modules.Publisher,
decisionengine.framework.logicengine, 75
decisionengine.framework.logicengine.BooleanExploreissiiommengine.framework.modules.Source,
   73
```

```
decisionengine.framework.modules.SourceProxy, decisionengine.framework.tests.SourceNOP,
decisionengine.framework.modules.tests,
                                               decisionengine.framework.tests.SourceWithSampleConfigN
decisionengine.framework.modules.tests.test_deddocisionengine.framework.tests.SupportsConfigPublisher
decisionengine.framework.modules.tests.test_EmphayEssinnovengine.framework.tests.test_client_errors,
decisionengine.framework.modules.tests.test_Modbedie;ionengine.framework.tests.test_client_server,
decisionengine.framework.modules.tests.test_mod legisteroemquinoersframework.tests.test_defaults,
decisionengine.framework.modules.tests.test_Pubbdishiem.engine.framework.tests.test_error_on_acquire,
decisionengine.framework.modules.tests.test_Sombreis;ionengine.framework.tests.test_module_program_opt
decisionengine.framework.modules.tests.test_Trahensifsnimm.engine.framework.tests.test_query_tool_server,
decisionengine.framework.modules.tests.test_tradersilsation.progrimmer.york.tests.test_reaper,
decisionengine.framework.modules.Transform,
                                               decisionengine.framework.tests.test_restart_channel,
decisionengine.framework.modules.translate_prodbucitsinammengine.framework.tests.test_sample_config,
decisionengine.framework.taskmanager, 86
                                               decisionengine.framework.tests.test_source_proxy,
decisionengine.framework.taskmanager.module_graph,94
                                               decisionengine.framework.tests.test_start_with_bad_cha
decisionengine.framework.taskmanager.ProcessingStake,
                                               decisionengine.framework.tests.test_start_with_no_char
decisionengine.framework.taskmanager.TaskManager, 94
                                               decisionengine.framework.tests.TransformNOP,
decisionengine.framework.tests, 94
decisionengine.framework.tests.ABTransform,
                                               decisionengine.framework.tests.TransformWithMissingPro
decisionengine.framework.tests.BATransform,
                                               decisionengine.framework.tests.WorkingSourceProxy,
decisionengine.framework.tests.ErrorOnAcquire,decisionengine.framework.util,96
                                               decisionengine.framework.util.fs, 94
decisionengine.framework.tests.FailingPublishedecisionengine.framework.util.metrics,95
                                               decisionengine.framework.util.reaper, 95
decisionengine.framework.tests.FailingSourceNOMPecisionengine.framework.util.singleton,
decisionengine.framework.tests.FailingSourceProbectyisionengine.framework.util.sockets,96
                                               decisionengine.framework.util.subclasses,
decisionengine.framework.tests.fixtures,
                                               decisionengine.framework.version, 96
decisionengine.framework.tests.ModuleProgramOpdeconsionengine.tests,97
                                               decisionengine.tests.test_framework_package,
decisionengine.framework.tests.PublisherNOP,
                                                   97
                                           Module
                                                          (class
                                                                                  decisio-
decisionengine.framework.tests.PublisherWithMissing@ginsenjework.modules.Module), 80
                                           ModuleProgramOptions
                                                                    (class
                                                                                  decisio-
decisionengine.framework.tests.SourceAlias,
                                                   nengine.framework.modules.describe), 83
                                           mydata()
                                                                     module
                                                                                   decisio-
```

		mework.log	gicengine.tests.	test_pandas	Z <b>.J%</b> CctD,E_DB_WI			module	decisio-	
	72	<i>(:</i>	11 .	1		ine.framewoi	<i>к.еп</i> діпе.	tests. <sub>Jixture</sub>	es),	
myengin		(in	module	decisio-	65	THE CONTRACT	· (:	1 1	,	
		ımework.10	gicengine.iesis.	test_cascaa	<i>e<b>E<u>Gr</u>dEs)</b></i> DB_WI			module	decisio-	
muonain	71	(in	modulo	dagigia		ine.framewoi			dagigio	
myengin		(in	module		PG_DE_DB_WI					etumos)
		ımework.io	gicengine.tests.	iesi_panaas	(_jaci), neng 47	ine.framewoi	к.аағаѕр	ice.aaiasoi	irces.iesis.ji	ciures),
myengin	72	(in	module	dagigia	PG_DE_DB_WI	TUOUT CCUE	MAC) (;	, modulo	danisia	
myengin		`			ith_negated <u>n</u> f <b>ng</b>					
	72	inie work.io	gicengine.iesis.	iesi_ruie_w	un_negaiea <u>i</u> g <b>a</b> g 55	ince.jramewor	к.иинагр	ice.iesis.jix	iures),	
myengin	. —	(in	module	decisio-	PG_DE_DB_WI	тноит сень	MA() (ii	n module	decisio-	
my crigin		*			_configurat <b>ion</b> )g					
	72	ine work.to	sicensine.iesis.	iesi_simpie_	_conjugurariang PG_DE_DB_WI'					
	7.2					ine.framewoi			accisio	
N					PG_PROG()	(in	тоа		decisio-	
name (dec	cisionengine	e.framework	k.dataspace.da	tasources.sa	lalchemy_d <b>xedk</b>	`				ctures).
riame (act	attribute),	•	space.aa	asources.sq	48	<u></u> geijeman10,51	**************************************			,
NotFoun			ework.engine.L	DecisionEng	iPGSPROGAD	(in	mod	lule	decisio-	
	attribute),					ine.framewor	rk.dataspe	ace.tests.fix	tures),	
NullDat		(class	in	decisio-	55	· ·	•			
	nengine.fra	*	taspace.dataso	urces.null),	PG_PROG()	(in	mod	lule	decisio-	
	50		•			ine.framewor	rk.engine.	tests.fixture	es),	
_					65					
O					PG_PROG()	(in	mod	lule	decisio-	
OFFLINE	(decisionen	gine.frame	work.taskmana	ger.Processi	ingState.Sta <b>t€</b> ng	ine.framewor	rk.tests.fix	tures), 90		
	attribute),				post_create	()			(decisio-	
orm_as_		(in	module	decisio-	_	ine.framewoi	rk.module	s.Source.S	ource	
	nengine.fra	ımework.da	taspace.dataso	urces.sqlalo	chemy_ds.ut <b>1181</b> /	(od), 81				
	43				post_create	()			(decisio-	
Ъ						ine.framewor	rk.module	s.SourceP1	oxy.SourceF	Proxy
Р						od), 81		_		
Paramet		(class	in	decisio-	Postgresql	(class		in .	decisio-	Τ.
	nengine.fra	ımework.me	odules.describe	2), 83		ine.framewor	rk.dataspo	ace.datasoi	ırces.postgre	esql),
Paramet		(class	in	decisio-	52	J 61 6	`		(1	
		ımework.me	odules.Publishe	er),	print_chann		-	CI IC	(decisio-	
	80						к.соппд.	CnanneiCo	ппднапагет	.ChannelConf
Paramet		(class	in	decisio-		(od), 35	· • • • • • • • • • • • • • • • • • • •	a a dada	docinio	
_			odules.Source),		print_consu		`	nodule	decisio-	
Paramet		(class	in	decisio-	neng 83	ine.framewoi	к.тоаше	s.prini_aes	scripiton),	
		ımework.me	odules.Transfor	m),	print_produ	cos() (	in n	nodule	decisio-	
	82		/· 1.1	,		ine.framewoi	`			
parse_p	rogram_op		(in module	decisio-	83	ine.jramewor	к.тошие	s.prini_aes	cripiton),	
		ımework.en	gine.DecisionE	ingine),	print_suppo	rted confi	a() (in	module	decisio-	
naccthn	68	iaunation	(in modul	a decisio		ine.framewor	_			
passenr			1() (in modul gicengine.Logi		83	ine.gramewer	ninio di lic	s.prini_acs	cripitoti),	
	74	imework.iog	gicengine.Logi	cEngine),	probably_ru	nnina()			(decisio-	
של אב א	/4 B_WITH_SC	HEMA()	(in module	decisio-			rk.taskma	nager.Proc	`	ProcessingSta
. G_DE_D			(in moaiite taspace.dataso		,	od), 84		5		
	47	ma work.uu	aaspace.aaaso	micco.icois.j	process_arg	* *			(decisio-	
PG DE D	B_WITH_SC	HEMA()	(in module	decisio-		ine.framewoi	rk.module	s.describe.	`	gramOptions
			taspace.tests.fi		~	od), 83				-
	55		r ====================================		ProcessingS	tate (	class	in	decisio-	

```
nengine.framework.taskmanager.ProcessingState)registry (decisionengine.framework.dataspace.datasources.sqlalchemy_a
                                                                                                                                            attribute), 41
produces() (decisionengine.framework.logicengine.Logic Reginest Handheine
                                                                                                                                                                            (class
                                                                                                                                                                                                                           decisio-
                                                                                                                                            nengine.framework.engine.DecisionEngine),
                    method), 74
produces()
                                            (in
                                                                  module
                                                                                                   decisio-
                                                                                                                                            68
                   nengine.framework.modules.Module), 80
                                                                                                                        required_keys
                                                                                                                                                                                                                          (decisio-
                                                                  module
                                                                                                   decisio-
                                                                                                                                            nengine.framework.dataspace.datablock.Header
produces()
                                            (in
                    nengine.framework.modules.Source), 81
                                                                                                                                            attribute), 60
produces()
                                            (in
                                                                  module
                                                                                                   decisio-
                                                                                                                        required_keys
                                                                                                                                                                                                                          (decisio-
                    nengine.framework.modules.Transform),
                                                                                                                                            nengine.framework.dataspace.datablock.Metadata
                                                                                                                                            attribute), 60
                                                                                                                        reset_connections()
ProductRetriever
                                                        (class
                                                                                 in
                                                                                                   decisio-
                                                                                                                                            nengine.framework.dataspace.datasource.DataSource
                   nengine.framework.dataspace.datablock),
                                                                                                                                            method), 63
publish() (decisionengine.framework.modules.Publisher.Pudsisteconnections()
                                                                                                                                                                                                                         (decisio-
                    method), 80
                                                                                                                                            nengine.framework.dataspace.datasources.null.NullDataSource
publish() (decisionengine.framework.tests.FailingPublisher.FailingPublisher.51
                                                                                                                        reset_connections()
                                                                                                                                                                                                                          (decisio-
publish() (decisionengine.framework.tests.PublisherNOP.PublisherNOP.publisher.framework.dataspace.datasources.postgresql.Postgresql
                                                                                                                                            method), 54
                    method), 88
                                           (class
                                                                                                   decisio-
Publisher
                                                                           in
                                                                                                                        reset_connections()
                                                                                                                                                                                                                         (decisio-
                   nengine.framework.modules.Publisher),
                                                                                                                                            nengine.framework.dataspace.datasources.sqlalchemy_ds.dataso
                    80
                                                                                                                                            method), 40
PublisherNOP
                                                 (class
                                                                                                   decisio-
                                                                                                                       reset_connections()
                                                                                                                                                                                                                         (decisio-
                                                                              in
                    nengine.framework.tests.PublisherNOP),
                                                                                                                                            nengine.framework.dataspace.datasources.sqlalchemy_ds.SQLAl
                                                                                                                                            method), 47
PublisherWithMissingConsumes (class in decisio- reset_source_flags()
                                                                                                                                                                                                                         (decisio-
                   nengine.framework.tests.PublisherWithMissingConsumes), nengine.framework.managers.ChannelManager.ChannelManager
                                                                                                                                            method), 75
put() (decisionengine.framework.dataspace.datablock.DataBlacention_interval
                                                                                                                                                                                                                          (decisio-
                    method), 60
                                                                                                                                            nengine.framework.dataspace.maintain.Reaper
                                                                                                                                            property), 64
R
                                                                                                                        rm_channel()
                                                                                                                                            nengine.framework.engine.DecisionEngine.DecisionEngine
reap() (decisionengine.framework.dataspace.maintain.Reaper
                                                                                                                                            method), 67
                    method), 64
                                                                                                                       rpc_block_while()
                                                                                                                                                                                                                          (decisio-
                                                                                                   decisio-
Reaper
                                      (class
                                                                        in
                                                                                                                                            nengine.framework.engine.DecisionEngine.DecisionEngine
                    nengine.framework.dataspace.maintain),
                                                                                                                                            method), 67
                   64
                                                                                                                     rpc_get_channel_log_level()
                                                                                                                                                                                                                          (decisio-
reaper()
                                                                module
                                                                                                   decisio-
                                                                                                                                            nengine.framework.engine.DecisionEngine.DecisionEngine
                   nengine.framework.dataspace.tests.test_Reaper),
                                                                                                                                            method), 67
                                                                                                 (decisio- rpc_get_log_level()
                                                                                                                                                                                                                         (decisio-
reaper_start()
                   nengine. framework. engine. Decision Engine. Decision Engine nengine. framework. engine. Decision Engine. Decision Engine nengine. The property of the prope
                                                                                                                                            method), 67
                   method), 67
                                                                                                 (decisio- rpc_kill_channel()
reaper_status()
                   nengine. framework. engine. Decision Engine. Decision Engine nengine. framework. engine. Decision Engine. Decision Engine nengine. The properties of the p
                                                                                                                                            method), 67
                   method), 67
                                                                                                                     rpc_paths (decisionengine.framework.engine.DecisionEngine.RequestHan
reaper_stop()
                                                                                                 (decisio-
                   nengine. framework. engine. Decision Engine. Decision Engine \ attribute), \ 68
                                                                                                                        rpc_print_product()
                                                                                                                                                                                                                          (decisio-
                   method), 67
                                                                                                                                            nengine.framework.engine.DecisionEngine.DecisionEngine
register_with_sources()
                                                                                                 (decisio-
                    nengine.framework.managers.ChannelManager.ChannelManuethod), 67
                                                                                                                        rpc_print_products()
                                                                                                                                                                                                                          (decisio-
                   method), 75
                                                                                                                                            nengine.framework.engine.DecisionEngine.DecisionEngine
```

method), 67				method), 77				
rpc_query_tool()		(decisio-	run()(de	ecisionengine.f	ramework.ma	nagers.Soi	urceSubscript	ionManager.
	k.engine.DecisionE						•	Ü
method), 67			run()(de	ecisionengine.f	ramework.tasi	kmanager.	TaskManager.	.TaskManag
<pre>rpc_reaper_start()</pre>		(decisio-		method), 85				
nengine.framewor	k.engine.DecisionE	ngine.Decis	si <b>ouE</b> n <b>}òg</b>	ic_engine()			(decisio-	
method), 67				nengine.frame	work.manager	rs.Channe	lManager.Ch	ınnelManag
<pre>rpc_reaper_status()</pre>		(decisio-		method), 76				
nengine.framewor	k.engine.DecisionE	Ingine.Decis	si <b>ouE</b> n <b>}og</b>				(decisio-	
method), 67				nengine.frame	work.taskman	ager.Task	Manager.Task	Manager
rpc_reaper_stop()		(decisio-		method), 85				
<u> </u>	k.engine.DecisionE	Ingine.Decis	si <b>out</b> E <b>ngido</b>				(decisio-	
method), 67				nengine.frame	work.managei	rs.Channe	lManager.Cho	ınnelManag
rpc_rm_channel()		(decisio-		method), 76				
	k.engine.DecisionE	Ingine.Decis	siounfi <u>ng</u> iolo				(decisio-	
method), 67	- 4			nengine.frame	work.taskman	ager.Task	Manager.Task	Manager
rpc_set_channel_log_l		(decisio-		method), 85				
- ·	k.engine.DecisionE	ingine.Decis	sioundingou		1 . 1		(decisio-	
method), 67				nengine.frame	work.taskman	ager.Taski	Manager.Task	Manager
rpc_show_config()		(decisio-		method), 85			(1	
<u> </u>	k.engine.DecisionE	ingine.Decis	si <b>ouh</b> in <del>gna</del>		7	CI.	(decisio-	73.4
method), 67		(1		nengine.frame	work.managei	rs.Channe	lManager.Cha	ınnelManag
<pre>rpc_show_de_config()</pre>	D	(decisio-		method), 76			(1	
<u> </u>	k.engine.DecisionE	ngine.Decis	si <b>oun</b> ngnæ				(decisio-	M
method), 67		(1:.:		nengine.frame	work.taskman	ager. 1 aski	Manager.1ask	Manager
<pre>rpc_start_channel()</pre>	de anaina DaaisianE	(decisio-	in Fution	method), 85			(decisio-	
method), 68	k.engine.DecisionE	ngine.Decis	приппадна	nengine.frame	work managa	rs Channa	`	annalManaa
rpc_start_channels()		(decisio-		method), 76	work.manager	rs.Cnanne	imanager.Cna	ınneımanag
_	k.engine.DecisionE	`	cinul utina				(decisio-	
method), 68	K.engine.DecisionE	ngine.Decis	приндера	nengine.frame	work taskman	ager Task	`	Manager
rpc_status()		(decisio-		method), 85	work.iaskman	uger.ruski	vianager. iask	Munuger
nengine framewor	k.engine.DecisionE		sionFnaina					
method), 68	K.engine.DecisionE	ngine.Decis	S	,				
<pre>rpc_stop() (decisionenging)</pre>	te.framework.engine	e.DecisionE	n <b>sëne D</b> GT	<i>id<u>o</u>rFe8ine_</i> ti	ime		(decisio-	
method), 68				nengine.frame	work.dataspa	ce.datasoi	irces.sqlalche	my_ds.db_se
<pre>rpc_stop_channel()</pre>		(decisio-		attribute), 42			_	
	k.engine.DecisionE	Ingine.Decis	si <b>gnEngi</b> ne	class (class	s in		decisio-	
method), 68				nengine.frame	work.dataspa	ce.datasoi	irces.sqlalche	my_ds.db_s
<pre>rpc_stop_channels()</pre>		(decisio-		43				
nengine.framewor	k.engine.DecisionE	Ingine.Decis	sign Remains	decisionengine.	.framework.do	ıtaspace.d	atasources.sq	lalchemy_ds
method), 68				attribute), 43				
Rule (class	in	decisio-	schema_	id(decisioneng	gine.framewor	k.dataspa	ce.datasource	s.sqlalchem
	k.logicengine.Rule)			attribute), 42				
<pre>rule_for() (decisionenging</pre>	e.framework.logice	ngine.FactL	ogkyyemtaac	<b>፤ፅ</b> ዊ <b>ଣ୍ଟୋ</b> Bioneng attribute), 43	gine.framewor	k.dataspa	ce.datasource	s.sqlalchem
RuleEngine (class	s in	decisio-	SconedS	ingleton	(class	in	decisio-	
- ,	k.logicengine.RuleI	Engine),	Scopeus	nengine.frame	*		accisio	
75	0 0	0 //	ScopedS	ingletonABC	(class	in	decisio-	
run() (decisionengine.fram	ework.engine.Work	ers.Worker	Jeopeus	nengine.frame	,			
method), 69			seconds	between rur	าร		(decisio-	
run() (decisionengine.fram	ework.managers.Cl	hannelMana	iger.Chann	elManaserame	work.dataspa	ce.maintai	in.Reaper	
method), 76	-			property), 64			T - '	
run() (decisionengine.fram	ework.managers.So	urceManag	er.SourceN					

```
send_data_product_to_subscribed()
                                                                                                                                                                         nengine.framework.managers.SourceManager),
                                                                                                                     (decisio-
                       nengine. framework. managers. Source Subscription Manager. Source Subscription Manager and the subscr
                                                                                                                                                 SourceSubscriptionManager (class in decisio-
                       method), 78
                                                                                                                                                                          nengine.framework.managers.SourceSubscriptionManager),
sequence_id
                                                                                                                      (decisio-
                       nengine.framework.dataspace.datasources.sqlalchemy_ds.db\u00e8schema.Taskmanager
                                                                                                                                                 SourceWithMissingProduces
                                                                                                                                                                                                                                     (class in decisio-
set() (decisionengine, framework, taskmanager, Processing State, P
                        method), 84
set_data_block()
                                                                                                                      (decisio-
                                                                                                                                                 SourceWithSampleConfigNOP
                                                                                                                                                                                                                                        (class
                                                                                                                                                                                                                                                             in
                                                                                                                                                                                                                                                                         decisio-
                       nengine.framework.modules.Module.Module
                                                                                                                                                                          nengine.framework.tests.SourceWithSampleConfigNOP),
                       method), 80
set_logging()
                                                                                    module
                                                                                                                        decisio-
                                                                                                                                                 spec_if_main()
                                                                                                                                                                                                               (in
                                                                                                                                                                                                                                      module
                                                                                                                                                                                                                                                                          decisio-
                                                           (in
                       nengine.framework.modules.de_logger),
                                                                                                                                                                          nengine.framework.modules.print description),
                        82
set_loglevel_value()
                                                                                                                     (decisio- SQLALCHEMY_PG_WITH_SCHEMA() (in module decisio-
                        nengine.framework.managers.ComponentManager.ComponentMinaframework.dataspace.datasources.tests.fixtures),
                       method), 77
                                                                                                                                                                          48
set_state()
                                                                                                                      (decisio- SQLALCHEMY_PG_WITH_SCHEMA() (in module decisio-
                       nengine.framework.dataspace.datablock.Metadata
                                                                                                                                                                          nengine.framework.dataspace.tests.fixtures), 55
                                                                                                                                                 SQLALCHEMY_PG_WITH_SCHEMA() (in module decisio-
                       method), 60
set_to_shutdown()
                                                                                                                      (decisio-
                                                                                                                                                                          nengine.framework.engine.tests.fixtures), 65
                       nengine.framework.taskmanager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.TaskManager.
                                                                                                                                                                          nengine.framework.tests.fixtures), 90
                       method), 86
                                                                                                                      (decisio- SQLALCHEMY_TEMPFILE_SQLITE() (in module decisio-
should_stop()
                       nengine.framework.taskmanager.ProcessingState.ProcessingState.framework.dataspace.datasources.tests.fixtures),
                       method), 84
SHUTDOWN (decisionengine.framework.taskmanager.ProcessEQISML@HPMMe_TEMPFILE_SQLITE() (in module decisio-
                                                                                                                                                                         nengine.framework.dataspace.tests.fixtures), 55
                        attribute), 85
shutdown() (decisionengine.framework.modules.Publisher SQLITE() (in module decisio-
                                                                                                                                                                          nengine.framework.engine.tests.fixtures), 65
                        method), 81
                                                                                                                     (decisio- SQLALCHEMY_TEMPFILE_SQLITE() (in module decisio-
SHUTTINGDOWN
                       nengine.framework.taskmanager.ProcessingState.State
                                                                                                                                                                          nengine.framework.tests.fixtures), 90
                                                                                                                                                 SQLA1chemyDS
                       attribute), 85
                                                                                                                                                                                                             (class
                                                                                                                                                                                                                                                                         decisio-
Singleton
                                                                                                                                                                          nengine.framework.dataspace.datasources.sqlalchemy_ds),
                                                     (class
                                                                                          in
                                                                                                                        decisio-
                        nengine.framework.util.singleton), 96
                                                                                                                                                                          44
SingletonABC
                                                           (class
                                                                                                                        decisio-
                                                                                                                                                 SOLA1chemvDS
                                                                                                                                                                                                             (class
                                                                                                                                                                                                                                                in
                                                                                                                                                                                                                                                                          decisio-
                       nengine.framework.util.singleton), 96
                                                                                                                                                                         nengine.framework.dataspace.datasources.sqlalchemy ds.dataso
sorted_rules()
                                                                                                                      (decisio-
                        nengine.framework.logicengine.FactLookup.FactLsvokup() (decisionengine.framework.dataspace.maintain.Reaper
                       method), 73
                                                                                                                                                                          method), 64
Source
                                              (class
                                                                                                                        decisio-
                                                                                                                                                 start_channel()
                                                                                                                                                                                                                                                                       (decisio-
                                                                                        in
                       nengine.framework.managers.SourceManager),
                                                                                                                                                                          nengine.framework.engine.DecisionEngine.DecisionEngine
                                                                                                                                                                          method), 68
Source
                                                                                                                                                 start_channels()
                                              (class
                                                                                        in
                                                                                                                        decisio-
                                                                                                                                                                                                                                                                       (decisio-
                        nengine.framework.modules.Source), 81
                                                                                                                                                                         nengine.framework.engine.DecisionEngine.DecisionEngine
SourceManager
                                                             (class
                                                                                                                        decisio-
                                                                                                                                                                          method), 68
                        nengine.framework.managers.SourceManager),
                                                                                                                                                start_sources()
                                                                                                                                                                                                                                                                        (decisio-
                                                                                                                                                                         nengine.framework.taskmanager.TaskManager.TaskManager
                        77
SourceNOP
                                                    (class
                                                                                           in
                                                                                                                        decisio-
                                                                                                                                                                          method), 86
                        nengine.framework.tests.SourceNOP), 89
                                                                                                                                                 State
                                                                                                                                                                                              (class
                                                                                                                                                                                                                                                                          decisio-
SourceProxy
                                                                                                                        decisio-
                                                                                                                                                                          nengine.framework.taskmanager.ProcessingState),
                                                         (class
                                                                                             in
                       nengine.framework.modules.SourceProxy),
                        81
                                                                                                                                                 state (decisionengine.framework.dataspace.datasources.sqlalchemy ds.db
                                                                                                                                                                          attribute), 42
SourceRunner
                                                           (class
                                                                                              in
                                                                                                                        decisio-
```

STEADY (decisionengine.framework.taskmanager.	Processin	- tate.State		
attribute), 85	_	ables ( <i>decisionengine.f</i>	ramework.dataspace.d	atasources.postgresql.Pos
stop() (decisionengine.framework.dataspace.mc method), 64	aintain.Re	er attribute), 54 ake_offline()	_	(decisio-
	(decisio-			(aecisio- ientManager.ComponentN
nengine.framework.engine.DecisionEng	gine.Decis	nEngine mathod) 77	ork.managers.Compon	ениманадет.Сотронени
method), 68	3	ask_dataproduct		(decisio-
	(decisio-	_		*
nengine.framework.engine.DecisionEng	gine.Decis	nengine.jramew nEngine attributa) 12	rork.aaiaspace.aaiasou	rces.sqlalchemy_ds.db_s
method), 68	3			(decisio
	decisio-	ask_header		(decisio-
nengine.framework.engine.DecisionEng			rork.aaiaspace.aaiasou	rces.sqlalchemy_ds.db_s
68	3 //	<pre>attribute), 43 ask_metadata</pre>		(decisio-
store_taskmanager() (	(decisio-			rces.sqlalchemy_ds.db_so
nengine.framework.dataspace.databloc	•	attribute), 43	ork.aaiaspace.aaiasou	rces.sqiaicnemy_as.ab_so
method), 60		unionic), 13	elass in	decisio-
	(decisio-	- '		rces.sqlalchemy_ds.db_so
nengine.framework.dataspace.datasour	•	rce 43	rork.aaiaspace.aaiasou	rces.sqiaicnemy_as.ab_so
method), 63		a clrManagon (a	olana in	decisio-
store_taskmanager() (	(decisio-	askrianayer (c	class in	Managan)
nengine.framework.dataspace.datasour	rces.null.N	nengine.jramev lDataSowrce	vork.taskmanager.1aski	vianager),
<pre>store_taskmanager()     nengine.framework.dataspace.datasour     method), 51</pre>		askmanager		(decisio-
	(decisio-	_		rces.sqlalchemy_ds.db_so
nengine.framework.dataspace.datasour		al.Postgresalbuta) 11	ork.aaiaspace.aaiasou	rces.sqiaicnemy_as.ab_so
method), 54	1 0	askmanager		(decisio-
	(decisio-			•
nengine.framework.dataspace.datasour	rces.sqlalc	my ds.datasource api.S	SQLAlchemyDS	rces.sqiaicnemy_as.ab_so
method), 40	1	askmanager		(decisio-
	(decisio-	askiiailayei	vork datasnace datasoi	(uecisio- uraes salalahamy de dh. si
store_taskmanager()  nengine.framework.dataspace.datasour	rces.sqlalc	my ds.SQLAlchemyDS	ork.aaiaspace.aaiasou	rces.sqiaicnemy_as.ab_so
method), 47	1	askmanager_id		(decisio-
	(decisio-			rces.sqlalchemy_ds.db_so
nengine.framework.dataspace.dataspac	ce.DataSp	e attribute), 41	701 K.auiuspuce.auiusou	rces.sqtatenenty_as.ao_st
method), 64	•	askmanager_id		(decisio-
Subscription (class in	decisio-	nanaina framay		rces.sqlalchemy_ds.db_sc
nengine.framework.managers.SourceSu	ıbscriptioi	lanager) attribute) 42	701 K.auiuspuce.auiusou	rces.sqtatenenty_as.ao_st
78	•	askmanager_id		(decisio-
Summary (class in	decisio-			rces.sqlalchemy_ds.db_sc
nengine.framework.util.metrics), 95		attribute), 43	701 K.aaiaspace.aaiasoa	rees.sqiaienemy_as.ao_se
<pre>supports_config() (in module</pre>	decisio-	askmanager_id		(decisio-
nengine.framework.modules.describe),	83			rces.sqlalchemy_ds.db_s
<pre>supports_config() (in module</pre>	decisio-	attribute), 43	701 K.aaiaspace.aaiasoa	rees.sqiaienemy_as.ao_se
nengine.framework.modules.Publisher)	),	askmanager_table		(decisio-
81		_	vork.dataspace.datasou	•
<pre>supports_config() (in module</pre>	decisio-	attribute), 63	701 K.auiuspuce.auiusou	rce.Datasource
nengine.framework.modules.Source), 8	1		lass in	decisio-
<pre>supports_config() (in module</pre>	decisio-		vork.managers.Channe	
nengine.framework.modules.Transform	),	76	.o.m.managers.Chaille	
82		, -	oine framework engine	DecisionEngine.StopState
	decisio-	attribute) 68		
nengine.framework.tests.SupportsConfi	gPublishe	est () (decisionenoine t	framework tests Moduli	ProgramOntions Acquire
89		method) 88	. a. ne work. resis.mount	S. amopuoiwa icquire

method), 88

method), 88

 $\verb|test()| (decision engine. framework. tests. Module Program Options. Acquire Works and the state of the st$ 

test() (decisionengine.framework.tests.ModuleProgramethod), 88	mOptions.Config:Teinplfatunework.config.tests.test_config), 33
test() (decisionengine.framework.tests.ModuleProgramenthod), 88	mOptients_10hannibed_module_missing_module()  (in module decisio-
	mOptions.Desneibeihligsamework.config.tests.test_config), 34
	mOp <b>น่องเร.Mhḥ</b> nnel_module_missing_parameters() (in module decisio-
test_acquire_for_sources() (in module decision nengine.framework.tests.test_module_program	nengine.framework.config.tests.test_config),
92	test_channel_names() (in module decisio-
test_allow_duplicate_source_proxy_keys()	nengine.framework.config.tests.test_config),
(in module decision	
nengine.framework.config.tests.test_de_std),	<pre>test_channel_no_config_files() (in module deci-</pre>
34	sionengine.framework.config.tests.test_config),
test_by_nonsense_is_err() (in module decision	
	ger),test_channel_no_modules() (in module decisio- nengine.framework.config.tests.test_config),
test_by_size() (in module decision	
nengine.framework.modules.tests.test_de_logg	ger),test_client_can_double_set_de_server_channel_log_level()
79	(in module decisio-
test_by_time() (in module decision	o- nengine.framework.tests.test_sample_config),
nengine.framework.modules.tests.test_de_logg	ger), 92
79	<pre>test_client_can_get_de_server_channel_config()</pre>
test_can_import() (in module decision	· ·
nengine.tests.test_framework_package), 97	nengine.framework.tests.test_sample_config),
test_change_port() (in module decision	
nengine.framework.engine.tests.test_startup),	test_client_can_get_de_server_channel_log_level()
66	(in module decisio-
test_channel_config_dir() (in module decision nengine.framework.config.tests.test_policies),	92
34	<pre>test_client_can_get_de_server_reaper_start_delay()</pre>
test_channel_empty_config() (in module decision	
nengine.framework.config.tests.test_config), 33	
test_channel_empty_dictionary() (in module dec	
sionengine.framework.config.tests.test_config)	
33	nengine.framework.tests.test_reaper), 92
test_channel_invalid_modules_list() (in module decision)	test_client_can_get_de_server_reaper_stop()
(in module decision nengine.framework.config.tests.test_config),	o- (in module decisio- nengine.framework.tests.test_reaper), 92
nengine.framework.config.tesis.tesi_config), 33	test_client_can_get_de_server_show_channel_logger_level()
test_channel_invalid_modules_no_keys()	(in module decisio-
(in module decisio	·
nengine.framework.config.tests.test_config),	test_client_can_get_de_server_show_config()
33	(in module decisio-
test_channel_invalid_modules_string()	nengine.framework.tests.test_sample_config),
(in module decision	ž , , , , , , , , , , , , , , , , , , ,
nengine.framework.config.tests.test_config),	<pre>test_client_can_get_de_server_show_logger_level()</pre>
33	(in module decisio-
test_channel_loading() (in module decision	·
nengine.framework.config.tests.test_config),	93
33	<pre>test_client_can_get_de_server_status()</pre>
test_channel_module_missing_all()	(in module decisio-
(in module decisio	o- nengine.framework.tests.test_sample_config),

9:	3		$nengine. framework. tests. test\_client\_errors),$
test_clie	nt_can_get_products()		91
(i		decisio-	<pre>test_client_de_config_is_json() (in module deci-</pre>
	engine.framework.tests.test_sample_c	config),	sionengine.framework.tests.test_defaults), 91
9:			test_client_err_returned_as_rc()
	nt_can_get_products_no_chann		(in module decisio-
(i		decisio-	nengine.framework.engine.tests.test_client_only),
	engine.framework.tests.test_sample_c	config),	66
9:			test_client_err_returned_as_rc()
	nt_can_get_products_no_chann		(in module decisio-
(i		decisio-	nengine.framework.engine.tests.test_query_tool_only)
	engine.framework.tests.test_start_with	h_bad_cha	
9.	•		test_client_err_returned_verbose_as_rc()
	nt_can_kill_one_channel()	,	(in module decisio-
(i		decisio-	nengine.framework.engine.tests.test_client_only),
	engine.framework.tests.test_sample_c	config),	66
9:		- ()	test_client_err_returned_verbose_as_rc()
	nt_can_kill_one_channel_forc		(in module decisio-
(i		decisio-	nengine.framework.engine.tests.test_query_tool_only)
no 9°	engine.framework.tests.test_sample_c	conjig),	66
	o nt_can_kill_one_channel_time	ou+ ( )	test_client_get_channel_log_fails_cleanly() (in module decisio-
		decisio-	
(i			nengine.framework.tests.test_sample_config), 93
9	engine.framework.tests.test_sample_c	conjig),	test_client_get_non_real_channel()
	nt_can_set_de_server_channel	log low	
(i		_10g_1ev decisio-	nengine.framework.tests.test_sample_config),
`	n mount engine.framework.tests.test_sample_c		93
9		orgig),	test_client_help() (in module decisio-
	nt_can_start_all_channel()		nengine.framework.engine.tests.test_client_only),
(i		decisio-	66
`	engine.framework.tests.test_sample_c		test_client_print_product() (in module decisio-
9:		3.07,	nengine.framework.tests.test_client_server), 91
test clie	nt_can_start_one_channel()		test_client_print_product_columns()
(i		decisio-	(in module decisio-
n	engine.framework.tests.test_sample_c	config),	nengine.framework.tests.test_client_server),
9:	-	J 0//	91
test_clie	nt_can_stop_channels()		<pre>test_client_print_product_columns_query()</pre>
(i		decisio-	(in module decisio-
n	engine.framework.tests.test_sample_c	config),	nengine.framework.tests.test_client_server),
9:	3		91
test_clie	nt_can_stop_one_channel()		<pre>test_client_print_product_json()</pre>
(i	n module	decisio-	(in module decisio-
n	engine.framework.tests.test_sample_c	config),	nengine.framework.tests.test_client_server),
9:	3		91
test_clie	nt_can_stop_server()(in module	e decisio-	<pre>test_client_print_product_not_real()</pre>
n	engine.framework.tests.test_sample_c	config),	(in module decisio-
9:			$nengine. framework. tests. test\_client\_server),$
test_clie	nt_cannot_double_start()		91
(i		decisio-	<pre>test_client_print_product_not_string()</pre>
	engine.framework.tests.test_sample_c	config),	(in module decisio-
9:			nengine.framework.tests.test_client_server),
	nt_cannot_wait_on_bad_state(		91
(i	n module	decisio-	test client print product query()

	(in	module	decisio-	- nengine.framework.dataspace.tests.test_datablock_zlib),
		ork.tests.test_client_	_server),	57
	91			test_conditional_fact() (in module decisio-
		oduct_types()		nengine.framework.logicengine.tests.test_fail_on_error),
	(in	module	decisio-	
		ork.tests.test_client_	_server),	test_config_templates() (in module decisio-
	91	aduat vantiaal(	`	nengine.framework.tests.test_module_program_options), 92
	.ent_print_pr ( <i>in</i>	oduct_vertical( module		7-
	(	moaute ork.tests.test_client_		(in module decisio-
	nengine.jramew 91	ork.iesis.iesi_ciieiii_	server),	nengine.framework.logicengine.tests.test_construction),
		nel_log_fails_c	leanlv()	71
	(in	module	decisio-	, -
	(	ork.tests.test_sample		(in module decisio-
	93	_ 1	_ 30//	nengine.framework.logicengine.tests.test_construction),
test_cli	.ent_start_no	n_real_channel(	)	71
	(in	module		- test_create_tables() (in module decisio-
i	nengine.framew	ork.tests.test_sample	e_config),	$nengine.framework.dataspace.datasources.tests.test\_datasource\_$
9	93			48
test_cli	.ent_status_m	sg_to_stdout()		<pre>test_DataBlock_constructor() (in module decisio-</pre>
	(in	module	decisio-	- · · ·
		ork.tests.test_client_	_server),	57
	91			test_DataBlock_duplicate() (in module decisio-
	_	_real_channel()		nengine.framework.dataspace.tests.test_datablock),
	(in	module	decisio-	
	nengine.framew 93	ork.tests.test_sample	e_conjig),	test_DataBlock_get_dataproducts() (in module decisio-
	93 .ent_wait_tim	eout works()		nengine.framework.dataspace.tests.test_datablock),
	(in	module	decisio-	
	(	ork.tests.test_sample		test_DataBlock_get_header() (in module decisio-
	93	orn.iesis.iesi_sampi		nengine.framework.dataspace.tests.test_datablock),
test_cli	ent_with_no_	.command_says_us	e_help()	57
		module		- test_DataBlock_get_metadata()(in module decisio-
ì	nengine.framew	ork.engine.tests.test		
	66	-		57
test_cli	.ent_with_no_	server() (in modi	ule decisio-	<pre>- test_DataBlock_get_taskmanager()</pre>
i	nengine.framew	ork.engine.tests.test_	_client_only),	
· ·	66			$nengine. framework. data space. tests. test\_datablock),$
		server_verbose(		57
	(in	module		- test_DataBlock_is_expired() (in module decisio-
		ork.engine.tests.test	_client_only),	
	66	-100 (* 1.1	,	57
	bine_one_lev	e1() (in moaui ork.config.tests.test_		<pre>- test_DataBlock_is_expired_with_key()</pre>
	nengine.jramew 34	ork.conjig.iesis.iesi_	_ae_sia),	(in module decisio- nengine.framework.dataspace.tests.test_datablock),
	-	el_skip_proxies	$\circ$	57
	(in	module	decisio-	
	`	ork.config.tests.test_		(in module decisio-
	34	orn.conjig.iesis.iesi_	_accstat);	nengine.framework.dataspace.tests.test_datablock),
	pound_fact_w	ith_spaces()		57
	(in	module	decisio-	- test_DataBlock_key_management_change_name()
	`	ork.logicengine.test		
	71	-		nengine.framework.dataspace.tests.test_datablock),
test_com	press()	(in module	decisio-	- 57

```
test_DataBlock_mark_expired() (in module decisio-test_fact_using_numpy_array() (in module decisio-
         nengine.framework.dataspace.tests.test_datablock),
                                                               nengine.framework.logicengine.tests.test_facts),
test_DataBlock_no_key_by_name()
                                                      test_fact_using_numpy_function()
                        module
                                            decisio-
                                                                               module
                                                                                                   decisio-
        nengine.framework.dataspace.tests.test_datablock),
                                                               nengine.framework.logicengine.tests.test facts),
test_DataBlock_to_str() (in module
                                            decisio- test_fact_with_fail_on_error()
         nengine.framework.dataspace.tests.test_datablock),
                                                               (in
                                                                               module
                                                                                                   decisio-
                                                               nengine.framework.logicengine.tests.test_facts),
         57
test_dataspace_config_finds_bad()
                                            decisio- test_fact_with_misspecified_attribute()
                        module
        nengine.framework.dataspace.tests.test_dataspace),
                                                                               module
                                                                                                   decisio-
                                                               (in
                                                               nengine.framework.logicengine.tests.test_fail_on_error),
         58
test_default_config()
                            (in
                                            decisio-
                                  module
         nengine.framework.engine.tests.test_startup),
                                                      test_fact_with_nested_names() (in module decisio-
                                                               nengine.framework.logicengine.tests.test_facts),
test_default_construction() (in module decisio-
        nengine.framework.logicengine.tests.test_constructiest,_fail_bad_config()
                                                                                   (in
                                                                                       module
                                                                                                  decisio-
                                                               nengine.framework.dataspace.tests.test Reaper),
test_delete_data_older_than_arg()
                                            decisio- test_fail_missing_config() (in module decisio-
        nengine.framework.dataspace.datasources.tests.test_datasourcegiapifcamework.dataspace.tests.test_Reaper),
                                 module
                                            decisio- test_fail_missing_config_key()
test_descriptions()
                          (in
        nengine.framework.tests.test_module_program_options),
                                                                               module
                                                                                                   decisio-
                                                               nengine.framework.dataspace.tests.test_Reaper),
test_duplicate_datablock() (in module decisio-
        nengine.framework.dataspace.datasources.tests.testestatidesaidecorageirror()
                                                                                 (in
                                                                                        module
                                                                                                  decisio-
                                                               nengine.framework.logicengine.tests.test_fail_on_error),
test_duplicate_datablock() (in module decisio-
         nengine.framework.dataspace.tests.test_dataspace@est_fail_small_retain() (in module decisio-
                                                               nengine.framework.dataspace.tests.test_Reaper),
test_duplicate_fact_names() (in module decisio-
         nengine.framework.logicengine.tests.test_duplicattesfstt_frainle.spmall_run_interval()
                                                                               module
                                                                                                   decisio-
                                                               (in
test_empty_config()
                          (in
                                 module
                                            decisio-
                                                               nengine.framework.dataspace.tests.test Reaper),
        nengine.framework.config.tests.test_config),
                                                      test_fail_start_two_reapers() (in module decisio-
test_empty_source_structure() (in module decisio-
                                                               nengine.framework.dataspace.tests.test_Reaper),
        nengine.framework.modules.tests.test EmptySource),
                                                      test_fail_wrong_config_key() (in module decisio-
                                                               nengine.framework.dataspace.tests.test Reaper),
test_error_conditions()
                             (in module
                                            decisio-
        nengine.framework.logicengine.tests.test_bool_function_name(),
                                                      test_false_fact_with_spaces() (in module decisio-
test_error_on_bad_names() (in module decisio-
                                                               nengine.framework.logicengine.tests.test_fail_on_error),
        nengine.framework.logicengine.tests.test_simple_configuration),
                                                      test_false_literal_fact() (in module decisio-
test_error_on_duplicate_keys() (in module deci-
                                                               nengine.framework.logicengine.tests.test_fail_on_error),
         sionengine.framework.config.tests.test_de_std),
                                                      test_generate_insert_query() (in module decisio-
test_exclusive_options() (in module decisio-
                                                               nengine.framework.dataspace.datasources.tests.test_postgresql),
         nengine.framework.engine.tests.test client only),
         66
                                                      test_get_datablock()
                                                                                 (in
                                                                                        module
                                                                                                   decisio-
```

```
nengine.framework.dataspace.datasources.tests.test_datasource_api),
                                                                                      test_get_last_generation_id_not_exist()
test_get_datablock()
                                                      module
                                                                       decisio-
                                                                                                                             module
                                                                                                     nengine.framework.dataspace.tests.test_dataspace),
              nengine.framework.dataspace.tests.test_dataspace),
test_get_dataproduct()
                                              (in
                                                      module
                                                                      decisio- test_get_metadata()
                                                                                                                                 (in
                                                                                                                                           module
                                                                                                                                                             decisio-
              nengine.framework.dataspace.datasources.tests.test_datasources_impisramework.dataspace.datasources.tests.test_datasource_
test_get_dataproduct()
                                              (in
                                                       module
                                                                       decisio- test_get_metadata()
                                                                                                                                 (in
                                                                                                                                            module
                                                                                                                                                             decisio-
              nengine.framework.dataspace.tests.test_dataspace),
                                                                                                    nengine.framework.dataspace.tests.test_dataspace),
test_get_dataproduct_not_exist()
                                                                                      test_get_metadata_not_exist() (in module decisio-
                                       module
                                                                       decisio-
                                                                                                    nengine.framework.dataspace.datasources.tests.test_datasource_
              (in
              nengine.framework.dataspace.datasources.tests.test_datasource_api),
                                                                                      test_get_metadata_not_exist() (in module decisio-
test_get_dataproduct_not_exist()
                                                                                                     nengine.framework.dataspace.tests.test_dataspace),
                                                                       decisio-
                                       module
              (in
              nengine.framework.dataspace.tests.test_dataspace@pest_get_taskmanager_exists() (in module decisio-
                                                                                                    nengine.framework.dataspace.datasources.tests.test_datasource_
test_get_dataproducts() (in module
              nengine.framework.dataspace.datasources.tests.testestatugerurtestatugerurtestatuger_exists() (in module decisio-
                                                                                                    nengine.framework.dataspace.tests.test_dataspace),
test_get_dataproducts() (in module
              nengine.framework.dataspace.tests.test dataspacetest_qet_taskmanager_not_exists()
                                                                                                    (in
                                                                                                                             module
                                                                                                                                                             decisio-
test_get_dataproducts_not_exist()
                                                                                                    nengine.framework.dataspace.datasources.tests.test_datasource_
                                       module
                                                                       decisio-
              nengine.framework.dataspace.datasources.tests.texe_stutgeturta_stpni)nager_not_exists()
                                                                                                                             module
                                                                                                                                                             decisio-
test_get_dataproducts_not_exist()
                                                                                                    nengine.framework.dataspace.tests.test_dataspace),
                                                                       decisio-
              nengine.framework.dataspace.tests.test_dataspace
pest_get_taskmanagers()
                                                                                                                                     (in module
                                                                                                                                                             decisio-
              58
                                                                                                    nengine.framework.dataspace.datasources.tests.test_datasource_
test_get_header()
                                        (in
                                                    module
                                                                       decisio-
              nengine.framework.dataspace.datasources.tests.testestatageources.dataspace.datasources.tests.testestatageources.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.dataspace.dataspace.dataspace.dataspace.dataspace.data
                                                                                                    nengine.framework.dataspace.tests.test_dataspace),
test_get_header()
                                        (in
                                                    module
                                                                       decisio-
              nengine.framework.dataspace.tests.test_dataspace)pest_get_taskmanagers_not_exist()
                                                                                                                              module
                                                                                                                                                             decisio-
test_get_header_not_exist() (in module decisio-
                                                                                                    nengine.framework.dataspace.datasources.tests.test_datasource_
              nengine.framework.dataspace.datasources.tests.test datasource api),
                                                                                      test_get_taskmanagers_not_exist()
test_get_header_not_exist() (in module decisio-
                                                                                                                              module
                                                                                                                                                             decisio-
                                                                                                    nengine.framework.dataspace.tests.test_dataspace),
              nengine.framework.dataspace.tests.test_dataspace),
test_get_last_generation_id() (in module decisio- test_global_channel_log_level_in_config()
              nengine.framework.dataspace.datasources.tests.test_datasou(ince_api),
                                                                                                                             module
                                                                                                    nengine.framework.tests.test_defaults), 91
test_get_last_generation_id() (in module decisio- test_global_config_dir() (in module decisio-
                                                                                                    nengine.framework.config.tests.test_policies),
              nengine.framework.dataspace.tests.test_dataspace),
test_get_last_generation_id_not_exist()
                                                                                      test_global_config_file() (in module decisio-
              (in
                                       module
                                                                       decisio-
                                                                                                    nengine.framework.config.tests.test_policies),
              nengine.framework.dataspace.datasources.tests.test_datasource_api),
```

test_	_has_config(	) (in	module	decisio-		78		
		ımework.date	aspace.datasoi	urces.tests.i	tett <u>e</u> sku <u>t</u> ansi	_	ct() (in module	
	49			7		nengine.framewo	rk.logicengine.tests.	test_fail_on_error),
test_	_has_config(		module	decisio-	<del></del>	72	(in modulo	doninio
	nengine.jrc 58	ітежоғк.аан	ispace.iesis.ie.	si_aaiaspac	ceyes t_mo	dule_alias()	(in module rk.tests.test_module_	decisio-
test	_has_methods	we expect	(in modul	e decisio-		92	rk.iesis.iesi_mounie_	_program_opiions),
		_			rceest_mo	dule_structure	e() (in module	decisio-
	57		1	_	// _		rk.modules.tests.test	_Module),
test_	_Header_cons					79		
	nengine.fra	amework.date	aspace.tests.te.	st_databloc	:ktest_mu	ltiple_consume	es_declarations(	)
	57					(· ·	module	decisio-
test_	_Header_is_v nengine.fra		in module aspace.tests.te.	decisio- st_databloo		79		_module_decorators),
	57	<i>(</i> )		7	test_mu		es_declarations(	
test_	_help()	`	module	decisio-	\	(· ·	module	decisio-
	92		s.test_module_			79		_module_decorators),
test_	_index_error nengine.fro 72		module icengine.tests.1		_		ture() (in module rk.modules.tests.test	
test_	_insert() nengine.fro 49	(in amework.date	module aspace.datasoi		_	nery_tool_csv() o <i>nnær<u>gi</u>npi</i> fframewo 92	(in module rk.tests.test_query_t	decisio- pol_server),
test_	_insert()	(in	module	decisio-	test_qu	ery_tool_defau	ılt() (in module	decisio-
	nengine.fra 58	ımework.datı	aspace.tests.te.	st_dataspa	ce),	nengine.framewo 92	rk.tests.test_query_t	ool_server),
test_	_jpath() nengine.fro 34	(in amework.con	module fig.tests.test_d		test_qu	nengine.framewo 66	() (in module rk.engine.tests.test_d	decisio- query_tool_only),
test_	_just_stop_n	o_error()	(in module	decisio-	test_qu	ery_tool_inval	.id_product()	
	nengine.fra	ımework.date	aspace.tests.te	st_Reaper)	,	(· ·	module	decisio-
	56	_					rk.tests.test_query_t	ool_server),
test_	_loop_of_sta			,		92		1
	(in nengine.fra 56	modul amework.date	e aspace.tests.te.		_	ery_tool_json( nengine.framewo 92	() (in module rk.tests.test_query_t	decisio- ool_server),
test_	_Metadata_co	nstructor	() (in module	e decisio-	test_qu	ery_tool_since	e() (in module	decisio-
	nengine.fra 57	ımework.datı	aspace.tests.te.	st_databloc	ck),	nengine.framewo 92	rk.tests.test_query_t	ool_server),
test_			*		_	ery_tool_with_		
	nengine.fra 57	ımework.datı	aspace.tests.te.	st_databloc	ck),	(· ·	module rk.engine.tests.test_c	decisio- query_tool_only),
test_	_minimal_jso					66		
	(in	modul	-	decisio-	test_qu		_no_server_verbo	
	34	·	fig.tests.test_c			(	module rk.engine.tests.test_o	decisio- query_tool_only),
test_	_minimal_jso	-				66		7
	(in nengine.fra 34	modui mework.con	le fig.tests.test_c	decisio- onfig),	test_re	_	ite() (in module rk.dataspace.tests.te	
test_		a_product_	_name_not_s	upported(	()test_re	aper_can_reap(	(in module	decisio-
	(in	modul		decisio-			rk.dataspace.tests.te	st_Reaper),
	nengine.fra	ımework.mo	dules.tests.test	EmptySou	rce).	56		

```
test_reset_connections() (in module decisio-
                                                              nengine.framework.dataspace.tests.test_Reaper),
        nengine.framework.dataspace.datasources.tests.test_datasource_api),
                                                     test_start_stop_stop()
                                                                                  (in
                                                                                       module
                                                               nengine.framework.dataspace.tests.test_Reaper),
test_restart_channel()
                                  module
                            (in
                                            decisio-
        nengine.framework.tests.test_restart_channel),
                                                     test_state_can_be_active() (in module decisio-
test_rule_that_does_not_fire()
                                                              nengine.framework.dataspace.tests.test Reaper),
                                            decisio-
        nengine.framework.logicengine.tests.test_cascadetlesttest_state_sets_timer_and_uses_it()
                                                                              module
                                                                                                  decisio-
test_rule_that_does_not_fire()
                                                               nengine.framework.dataspace.tests.test_Reaper),
                        module
                                            decisio-
        nengine.framework.logicengine.tests.test_pandas_fast__stop_failing_source_proxy()
                                                                              module
                                                                                                  decisio-
test_rule_that_does_not_fire()
                                                               nengine.framework.tests.test_source_proxy),
                        module
                                            decisio-
                                                               94
        nengine.framework.logicengine.tests.test_rule_withestgated_factaskmanager() (in module decisio-
                                                              nengine.framework.dataspace.datasources.tests.test_datasource_
test_rule_that_does_not_fire()
                        module
                                            decisio- test_store_taskmanager() (in module decisio-
        nengine.framework.logicengine.tests.test_simple_configurationgine.framework.dataspace.tests.test_dataspace),
test_rule_that_fires()
                            (in
                                  module
                                            decisio- test_supports_config()
                                                                                  (in
                                                                                        module
                                                                                                  decisio-
        nengine.framework.logicengine.tests.test cascaded rules), nengine.framework.modules.tests.test module decorators),
                                            decisio- test_syntax_error()
test_rule_that_fires()
                           (in
                                  module
                                                                                (in
                                                                                       module
                                                                                                  decisio-
        nengine. framework. logic engine. tests. test\_pandas\_fact),
                                                              nengine.framework.logicengine.tests.test_facts),
                                            decisio- test_syntax_error_in_config_names_bad_file()
test_rule_that_fires()
                            (in
                                  module
        nengine.framework.logicengine.tests.test_rule_with_negated_ifact),
                                                                              module
                                                                                                  decisio-
                                                               nengine.framework.config.tests.test_config),
test_rule_that_fires()
                            (in
                                  module
                                            decisio-
        nengine.framework.logicengine.tests.test_simple_configuration_structure() (in module decisio-
                                                               nengine.framework.modules.tests.test_Transform),
test_simple_fact()
                          (in
                                 module
                                            decisio-
        nengine.framework.logicengine.tests.test_facts), test_translate_all()
                                                                                 (in
                                                                                       module
                                                                                                  decisio-
                                                              nengine.framework.modules.tests.test translate product name),
test_source_fail_can_be_fixed()
                        module
                                            decisio- test_translate_illegal_characters()
        nengine.framework.dataspace.tests.test_Reaper),
                                                                              module
                                                                                                  decisio-
                                                              nengine.framework.modules.tests.test translate product name),
test_source_only_channel() (in module decisio-
        nengine.framework.tests.test_error_on_acquire), test_translate_none()
                                                                                  (in
                                                                                        module
                                                                                                  decisio-
                                                              nengine.framework.modules.tests.test_translate_product_name),
test_source_structure() (in module decisio-
        nengine.framework.modules.tests.test_Source),
                                                     test_translate_simple() (in module
                                                                                                  decisio-
                                                              nengine.framework.modules.tests.test_translate_product_name),
test_start_delay()
                          (in
                                 module
                                            decisio-
        nengine.framework.dataspace.tests.test_Reaper), test_translate_with_underscores()
                                                                              module
                                                                                                  decisio-
test_start_from_nothing() (in module decisio-
                                                               nengine.framework.modules.tests.test_translate_product_name),
        nengine.framework.tests.test_start_with_no_channels),
                                                     test_trivial_configuration() (in module decisio-
                                                              nengine.framework.logicengine.tests.test construction),
test_start_stop()
                         (in
                                module
                                            decisio-
```

```
71
                                                                                                                                         nengine.framework.tests.TransformNOP),
test_true_fact()
                                                                      module
                                                                                                 decisio-
                                                     (in
                   nengine.framework.logicengine.tests.test_fail_on_\mathbb{Trams}.formWithMissingProducesConsumes
                                                                                                                                         (class
                                                                                                                                                                                                                       decisio-
test_true_literal_fact() (in module decisio-
                                                                                                                                         nengine.framework.tests.TransformWithMissingProducesConsum
                   nengine.framework.logicengine.tests.test fail on error),
                   72
                                                                                                                                                                                        module
                                                                                                                                                                                                                       decisio-
                                                                                                                      translate()
                                                                                                                                                                   (in
                                                                                                                                         nengine.framework.modules.translate_product_name),
test_update()
                                                (in
                                                                    module
                                                                                                 decisio-
                   nengine.framework.dataspace.datasources.tests.test_datasource_api),
                                                                                                                      translate_all()
                                                                                                                                                                         (in
                                                                                                                                                                                           module
                                                                                                                                                                                                                       decisio-
test_update()
                                                                    module
                                                                                                 decisio-
                                                                                                                                         nengine.framework.modules.translate_product_name),
                   nengine.framework.dataspace.tests.test_dataspace),
                                                                                                 decisio-
test_update_bad()
                                                                       module
                                                       (in
                   nengine.framework.dataspace.datasources.tests.testngdataxdedce.cquis,s()
                                                                                                                                                                                                                     (decisio-
                   49
                                                                                                                                         nengine.framework.engine.Workers.Workers
test_update_bad()
                                                       (in
                                                                       module
                                                                                                 decisio-
                                                                                                                                         method), 69
                   nengine.framework.dataspace.tests.test_dataspace@pdate() (decisionengine.framework.dataspace.datasource.DataSource
                                                                                                                                         method), 63
test_valid_but_empty_config() (in module decisio-
                                                                                                                     update() (decisionengine.framework.dataspace.datasources.null.NullData
                   nengine.framework.config.tests.test_config), 34
                                                                                                                                         method), 52
test_valid_dir()
                                                                      module
                                                                                                 decisio-
                                                     (in
                                                                                                                     update() (decisionengine.framework.dataspace.datasources.postgresql.Po
                   nengine.framework.config.tests.test_policies),
                                                                                                                                          method), 54
                                                                                                                     update() (decisionengine.framework.dataspace.datasources.sqlalchemy_a
test_working_source_proxy() (in module decisio-
                                                                                                                                         method), 40
                   nengine.framework.tests.test source proxy), 94
                                                                                                                     update() (decisionengine.framework.dataspace.datasources.sqlalchemy_a
test_wrong_configuration() (in module decisio-
                                                                                                                                          method), 47
                   nengine.framework.logicengine.tests.test_constructionate() (decisionengine.framework.dataspace.dataspace.DataSpace
                                                                                                                                         method), 64
test_wrong_product_names() (in module decisio-
                   nengine.framework.modules.tests.test\_module\_dec  variable fractions variable vari
                                                                                                                      valid_dir()
                                                                                                                                                                  (in
                                                                                                                                                                                        module
                                                                                                                                                                                                                       decisio-
test_wrong_product_types() (in module decisio-
                                                                                                                                          nengine.framework.config.policies), 36
                   nengine.framework.modules.tests.test_module_decorprorsstates
                                                                                                                                                                                                                      (decisio-
                   79
                                                                                                                                          nengine.framework.dataspace.datablock.Metadata
test_zdumps()
                                                                    module
                                                                                                 decisio-
                                                (in
                                                                                                                                          attribute), 60
                   nengine.framework.dataspace.tests.test_datablockvafibaconfig
                                                                                                                                                                   (class
                                                                                                                                                                                                                       decisio-
                   57
                                                                                                                                          nengine.framework.config.ValidConfig),
test_zloads()
                                                (in
                                                                    module
                                                                                                 decisio-
                   nengine. framework. dataspace. tests\_datablock_{\coloredge{table}}{\coloredge{table}} (decisionengine. framework. dataspace. datasources. sqlalchemy\_ds. ddts. dataspace. datasources. sqlalchemy\_ds. ddts. dataspace. dat
                                                                                                                                          attribute), 41
Transform
                                          (class
                                                                          in
                                                                                                 decisio-
                                                                                                                     verify_products()
                                                                                                                                                                            (in
                                                                                                                                                                                             module
                                                                                                                                                                                                                       decisio-
                   nengine.framework.modules.Transform),
                                                                                                                                         nengine.framework.modules.Module), 80
                   82
                                                                                               (decisio- W
transform()
                   nengine. framework. modules. Transform. Transform\\ \textbf{wait\_for\_all()}
                                                                                                                                                                                                                     (decisio-
                   method), 82
                                                                                                                                          nengine.framework.managers.ChannelManager.ChannelManager
transform()
                                                                                               (decisio-
                                                                                                                                         method), 76
                   nengine.framework.tests.TransformNOP.TransformNOP_walt_for_all()
                                                                                                                                                                                                                     (decisio-
                                                                                                                                          nengine.framework.taskmanager.TaskManager.TaskManager
transform()
                                                                                               (decisio-
                   nengine.framework.tests.TransformWithMissingProducesConsumes.TransformWithMissingProducesConsumes wall_for_all_sources()
                   method), 90
                                                                                                                                         nengine.framework.managers.ChannelManager.ChannelManager
TransformNOP
                                                (class
                                                                                                 decisio-
                                                                                                                                         method), 76
```

```
wait_for_any()
                                            (decisio-
        nengine.framework.taskmanager.TaskManager.TaskManager
        method), 86
wait_for_any_source()
                                            (decisio-
        nengine.framework.managers.ChannelManager.ChannelManager
        method), 76
wait_for_registration()
                                            (decisio-
        nengine.framework.managers.ChannelManager.ChannelManager
        method), 76
wait_until()
                                            (decisio-
        nengine.framework.engine.Workers.Worker
        method), 69
wait_until()
                                            (decisio-
        nengine.framework.taskmanager.ProcessingState.ProcessingState
        method), 84
wait_while()
                                            (decisio-
        nengine.framework.engine.Workers.Worker
        method), 69
wait_while()
                                            (decisio-
        nengine. framework. task manager. Processing State. Processing State\\
        method), 84
Worker
                 (class
                                 in
                                             decisio-
        nengine.framework.engine.Workers), 69
Worker
                 (class
                                             decisio-
        nengine.framework.taskmanager.TaskManager),
Workers
                  (class
                                 in
                                             decisio-
        nengine.framework.engine.Workers), 69
Workers.Access
                       (class
                                             decisio-
                                    in
        nengine.framework.engine.Workers), 69
WorkingSourceProxy
                           (class
                                             decisio-
        nengine.framework.tests.WorkingSourceProxy),
        90
Ζ
zdumps()
                             module
                                             decisio-
                  (in
        nengine.framework.dataspace.datablock),
zloads()
                             module
                                             decisio-
                  (in
        nengine.framework.dataspace.datablock),
```