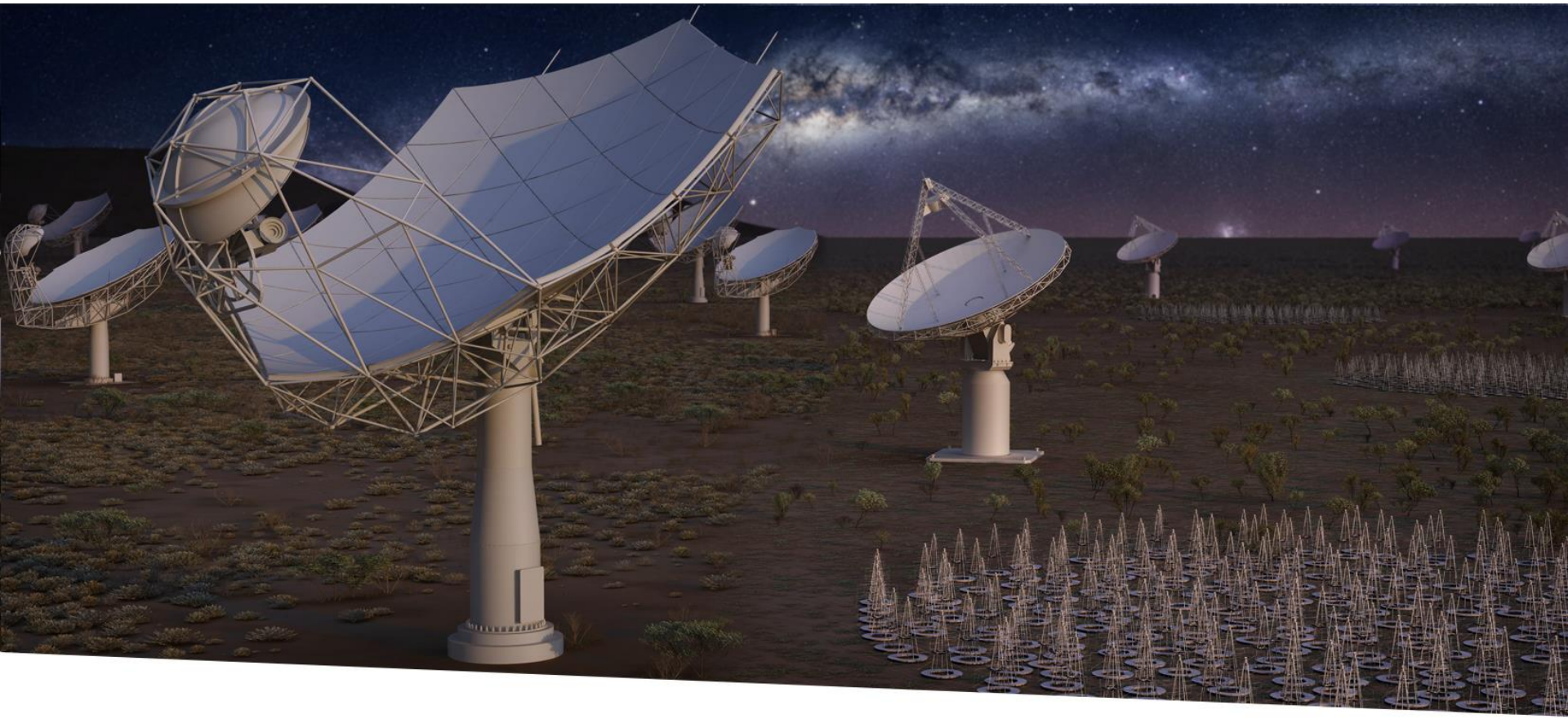
rucio-analysis



SQUARE KILOMETRE ARRAY

Exploring the Universe with the world's largest radio telescope

Rob Barnsley

21/10/2020

Introduction

rucio-analysis: Extensible python3 framework for repeated stress testing of the ESCAPE datalake.

Currently used in the site-to-site replication testing. This can be seen in the SKA_SKAO_BARNSLEY-testing scope accessible from the rucio-events dashboard.

Task cronned & running hourly on an STFC cloud hosted VM.

Future uses include mocking prospective SKA data workflows (w/QoS) in the datalake.

Workflow (1/4)

Requirements for a test:

- A new task yaml configuration file linking test logic module/class and any required input parameters
- Logic for the test
- (optional, if to be cronned) A wrapper script to run as a cronjob

Workflow (2/4)

creating a new task config

1. Create new test configuration file in etc/, defining new test module location (relative to src/), <module_name>, & class name, <class_name>. There's a stub for this in /etc/tests.stub.yaml
2. In configuration file, add any keyword arguments <kwargs> to be sent to the test class

e.g. /etc/tests.stub.yml

1.
2.

```
1  test-stub:
2      description: "Test stub"
3      module_name: "tests.stub"
4      class_name: "TestStub"
5      enabled: false
6      args:
7      kwargs:
8        text: "Hello World!"
```

Workflow (3/4)

writing the test logic

3. Copy the test stub class, src/tests/stub.py, and place it in the previously defined location (usually src/tests/)
4. Assign previous kwargs in run() constructor to access them
5. Write test logic in placeholder

e.g. /src/tests/stub.py

```
1  from tests import Test
2
3  class TestStub(Test):
4      """ Stub test class.
5      """
6      def __init__(self, logger):
7          super().__init__(logger)
8
9
```

```
10     def run(self, args, kwargs):
11         super().run()
12         self.tic()
13         try:
14             # Assign variables from test.yml kwargs.
15             #
16             text = kwargs['text']                    4.
17
18         except KeyError as e:
19             self.logger.critical("Could not find necessary kwarg for test.")
20             self.logger.critical(repr(e))
21             exit()
22
23         # Your code here.
24         self.logger.info(text)                       5.
25
26         self.toc()
27         self.logger.info("Finished in {}s".format(
28             round(self.elapsed)))
```

Workflow (4/4)
output

```
[user@b802f5113379 /]$ voms-proxy-init --cert /opt/rucio/etc/client.crt --key
/opt/rucio/etc/client.key --voms escape
[user@b802f5113379 /]$ cd ~/rucio-analysis/src/
[user@b802f5113379 src]$ python3 run-analysis.py -v -t ../etc/tests.stub.yml

2020-09-16 14:35:29,835 INFO    Parsing tests file
2020-09-16 14:35:29,865 DEBUG   Constructing instance of TestStub()
2020-09-16 14:35:29,873 INFO    Executing TestStub.run()
2020-09-16 14:35:29,883 INFO    Hello World!
2020-09-16 14:35:29,950 INFO    Finished in 0s
2020-09-16 14:35:29,970 DEBUG   Deconstructing instance of TestStub()
```

Other points

- Polymorphic interface to both API and CLI Rucio functionality in src/rucio_wrappers.py.

```
10   class RucioWrappers:
60   class RucioWrappersCLI(RucioWrappers):
159  class RucioWrappersAPI(RucioWrappers):
```

- Cronnable, launching within rucio-client-container dockerised environment e.g.
  1. Create a new job script in etc/cron/crontab/jobs (see etc/cron/jobs/test.sh for an example)
  2. Add entry for this script in etc/cron/crontab
  3. Execute etc/install-crontab.sh (overwrites existing user crontab!)

Current site-to-site replication test

Transfer Matrix: transfer-done/transfer-submitted

src \ dst	DESY-DCACHE	SARA-DCACHE	PIC-DCACHE	EULAKE-1	LAPP-DCACHE	IN2P3-CC-DCACHE	CNAF-STORM	ALPAMED-DPM	GSI-ROOT	INFN-NA-DPM	LAPP-WEBDAV
DESY-DCACHE	NO DATA	100%	50%	100%	100%	100%	100%	100%	100%	100%	100%
SARA-DCACHE	92%	NO DATA	92%	92%	92%	92%	92%	75%	92%	92%	92%
PIC-DCACHE	92%	92%	NO DATA	92%	92%	92%	77%	75%	92%	92%	92%
EULAKE-1	NO DATA	NO DATA	100%	NO DATA	NO DATA	NO DATA	100%	NO DATA	NO DATA	NO DATA	NO DATA
LAPP-DCACHE	NO DATA	NO DATA	NO DATA	NO DATA	NO DATA	NO DATA	50%	100%	NO DATA	NO DATA	NO DATA
IN2P3-CC-DCACHE	92%	92%	100%	92%	92%	NO DATA	86%	92%	92%	92%	92%
CNAF-STORM	100%	100%	100%	100%	100%	100%	NO DATA	100%	100%	NO DATA	92%
ALPAMED-DPM	100%	100%	82%	100%	91%	100%	85%	NO DATA	91%	100%	91%
GSI-ROOT	92%	92%	92%	92%	92%	92%	92%	69%	NO DATA	92%	92%
INFN-NA-DPM	86%	86%	88%	86%	NO DATA	86%	NO DATA	89%	71%	NO DATA	NO DATA
LAPP-WEBDAV	NO DATA	NO DATA	0%	NO DATA	NO DATA	NO DATA	0%	NO DATA	NO DATA	NO DATA	NO DATA

```
21          - PIC-DCACHE
22          - SARA-DCACHE
23          - LAPP-WEBDAV
24        scope: SKA_SKAO_BARNSLEY-testing
```

IE
:1984cd3ed014f94633

estReplication()

Remarks

Always open to contributions!

https://github.com/ESCAPE-WP2/rucio-analysis
https://hub.docker.com/repository/docker/projectescape/rucio-analysis

SQUARE KILOMETRE ARRAY

Exploring the Universe with the world's largest radio telescope