National Research Foundation | SARAO South African Radio Astronomy Observatory

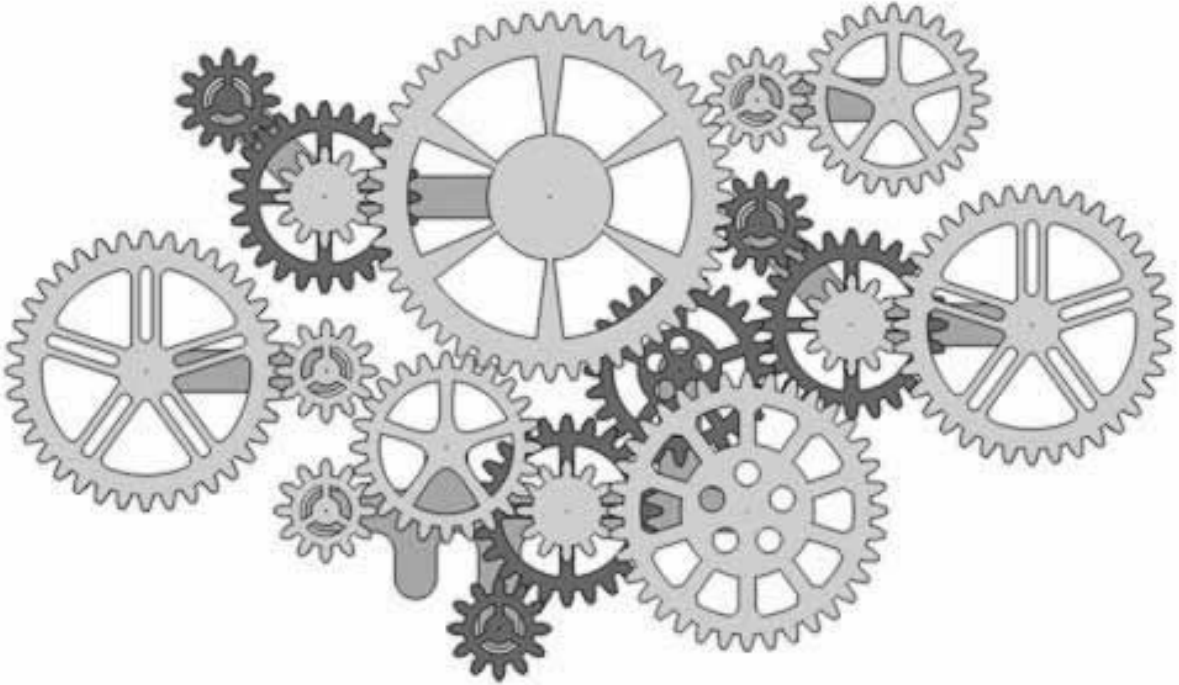
ICALEPCS

Logging and Tracing in the SKA Telescope Control System

Samuel Twum, Junior Software Engineer (SARAO)
and W. Bode, A. F. Joubert, K. Madisa, P.S. Swart, A. J. Venter, A.Bridger

19th October, 2021 @ ICALEPCS 2021, Software Technology Evolution Track

Preview to Theory



Crossing the lines from being clueless to having full understanding of your system's internal state is a function of the observability of your distributed system

Image credits in order of left to right:
https://www.giftstogive.org/lightbulb-1/
https://www.gearswalldecor.com/
https://www.pngwing.com/en/free-png-tattv

The 3 Pillars of Observability

Logs

records of activity within the system

Metrics

measurement of various activities in a system

Tracing

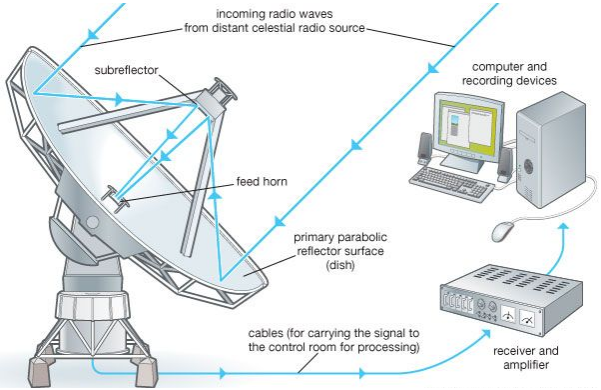
the path taken by a request as it moves through a distributed system

Observability: inferring the internal state of a system from its external outputs

~

Cloud Native Observability for DevOps Teams by J. Heather (2021)

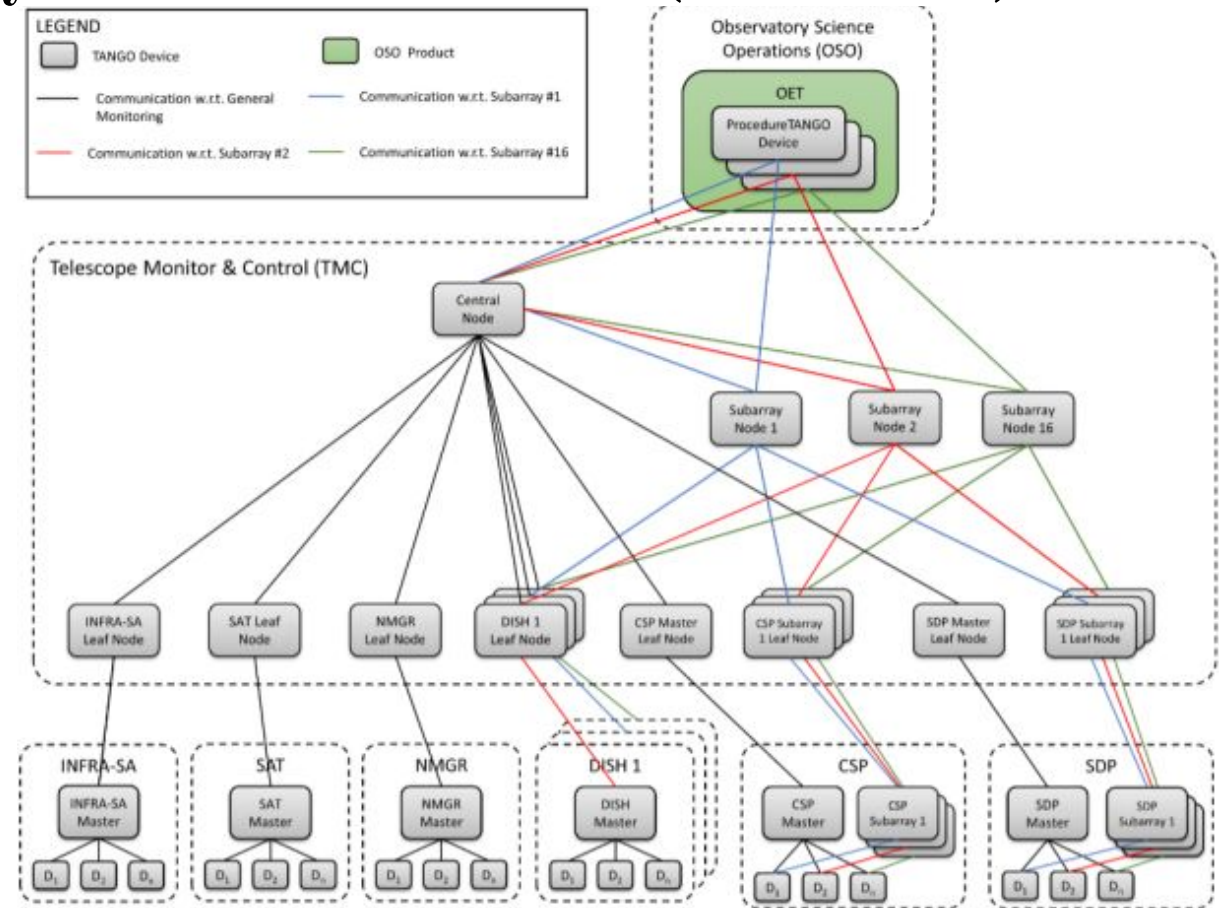
The SKA Control System Architecture (Overview)



In a typical radio telescope, a large parabolic antenna, or dish, collects incoming radio waves and focuses them onto a smaller antenna called the feed horn. The signal is then carried to the radio receiver.

~

Encyclopedia Britannica, 2010



Hierarchy of tango devices demonstrating the control of the Mid telescopes

Road Map to Tracing Solution

- Harmonised logging to provide consistently formatted logs across all applications
- Provisioned a unique ID generator to be used as tags in the logs
- Provisioned a context handler that emits ska formatted logs with a transaction ID injected

SKA Log Message Format

SKA log message:

```
VERSION "|" TIMESTAMP "|" SEVERITY "|" [THREAD-ID] "|" [FUNCTION] "|" [LINE-LOC] "|" [TAGS] "|" MESSAGE
LF
```

Examples:

```
1|2019-12-31T23:12:37.526Z|INFO||testpackage.testmodule.TestDevice.test_fn|test.py#1|tango-device:my/dev/name| Regular
information should be logged like this FYI
1|2019-12-31T23:45:42.328Z|DEBUG||testpackage.testmodule.TestDevice.test_fn|test.py#150|| x = 67, y = 24
1|2019-12-31T23:49:53.543Z|WARNING||testpackage.testmodule.TestDevice.test_fn|test.py#16|| z is unspecified, defaulting to 0!
1|2019-12-31T23:50:17.124Z|ERROR||testpackage.testmodule.TestDevice.test_fn|test.py#165|site:Element| Could not connect to
database!
1|2019-12-31T23:51:23.036Z|CRITICAL||testpackage.testmodule.TestDevice.test_fn|test.py#16|| Invalid operation. Cannot continue.
```

The Log Message Standard is not an extension of syslog/RFC5234 format. Read the SKA Developer Portal for more details

The SKA Logging Configuration Library

```
import logging
from ska_ser_logging import configure_logging

def main():
    configure_logging()
    logger = logging.getLogger("ska.example")
    logger.info("Logging started for Example application")
main()
```

```
1|2021-10-12T21:58:41.222Z|INFO|MainThread|main|<ipython-input-4-13a3535fcc78>#4||Logging started for
Example application
```

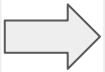
SKA Unique Identifier (SKUID) Library

```
import os

from ska_ser_skuid.client import SkuidClient

def get_transaction_id():
    if "SKUID_URL" in os.environ and os.environ["SKUID_URL"]:
        client = SkuidClient(os.environ["SKUID_URL"])
        return client.fetch_transaction_id()
    return SkuidClient.get_local_transaction_id()

transaction_id = get_transaction_id()
print(transaction_id)
```

```
txn-local-20211012-643313143
                OR
txn-t0001-20200914-123456789
```

SKA Log Transactions Library

```
with transaction('My Command') as
transaction_id:
    # do stuff
    ...
```



```
from ska_ser_log_transactions  import transaction

def command(self, parameter_json):
   parameters = json.reads(parameter_json)
   with transaction('My Command', parameters) as
transaction_id:
       # ...
       parameters['transaction_id'] = transaction_id
       device.further_command(json.dumps( parameters))
       # ...
```

Transaction Message Formats:

- On Entry:
  - Transaction[id]: Enter[name] with parameters [arguments] marker[marker]
- On Exit:
  - Transaction[id]: Exit[name] marker[marker]
- On Exception
  - Transaction[id]: Exception[name] marker[marker] -- Stacktrace --

The marker can be used to match entry/exception/exit log messages.

SKA Log Transactions Library

Example ska formatted logs for successful transaction:

```
1|2020-10-01T12:49:31.119Z|INFO|Thread-210|log_entry|transactions.py #154||Transaction[txn-local-20201001-981667980]: Enter[Command] with parameters [{}] marker[52764]
```

```
1|2020-10-01T12:49:31.129Z|INFO|Thread-210|log_exit|transactions.py #154||Transaction[txn-local-20201001-981667980]: Exit[Command] marker[52764]
```

Example ska formatted logs for failed transaction:

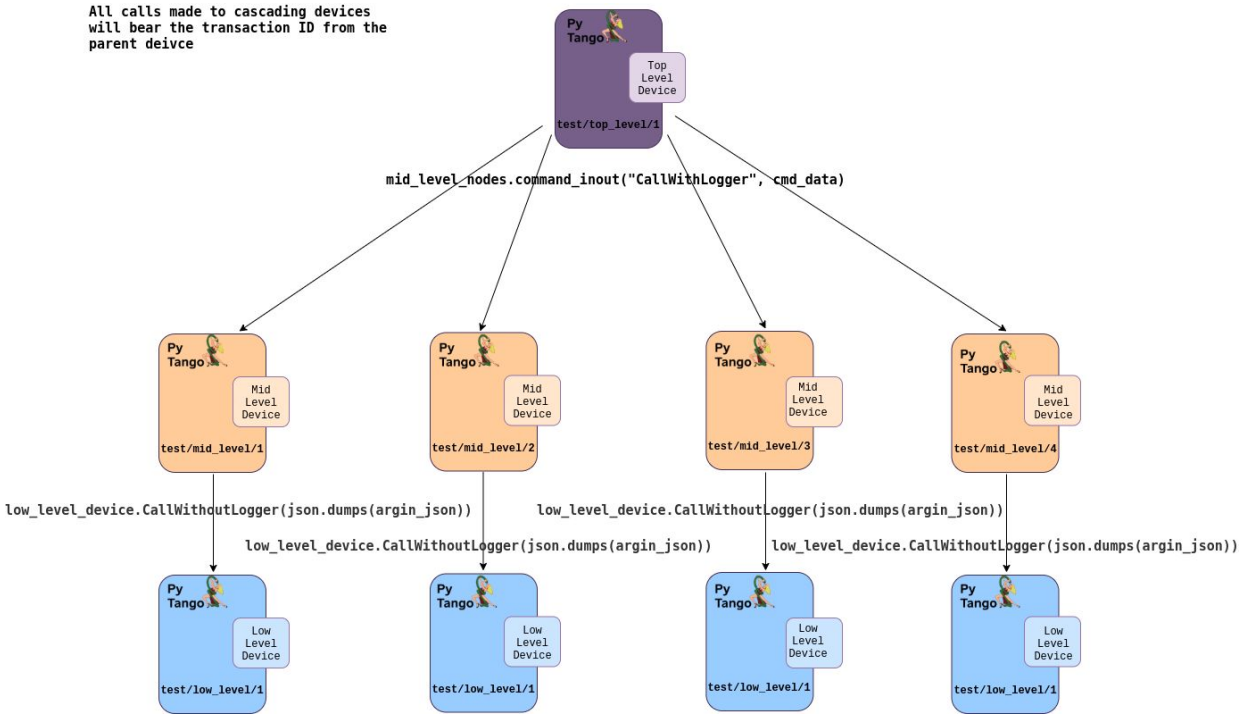
```
1|2020-10-01T12:51:35.588Z|INFO|Thread-204|log_entry|transactions.py #154||Transaction[txn-local-20201001-354400050]: Enter[Transaction thread [7]] with parameters [{}] marker[21454]
```

```
1|2020-10-01T12:51:35.598Z|ERROR|Thread-204|log_exit|transactions.py #149||Transaction[txn-local-20201001-354400050]: Exception[Transaction thread [7]] marker[21454]
Traceback (most recent call last):
 File "python_file.py", line 27, in thread_with_transaction_exception
   raise RuntimeError("An exception has occurred")
RuntimeError: An exception has occurred
```

```
1|2020-10-01T12:51:35.601Z|INFO|Thread-204|log_exit|transactions.py #154||Transaction[txn-local-20201001-354400050]: Exit[Transaction thread [7]] marker[21454]
```

It does not support a multithreaded case at the moment

Log Transaction Illustration in Tango Example



Tango-example: a project that demonstrates how to structure an SKA project that provides some simple Tango devices coded in PyTango.

Closing

- Future work
  - Build the tooling we need to visualise the traces from this working iteration
- Conclusion
  - To fully understand a distributed system requires distributed tracing
  - All the work done are available in public repositories under the ska-telescope organisation in: ska-ser-logging, ska-ser-skuid and ska-ser-log-transactions