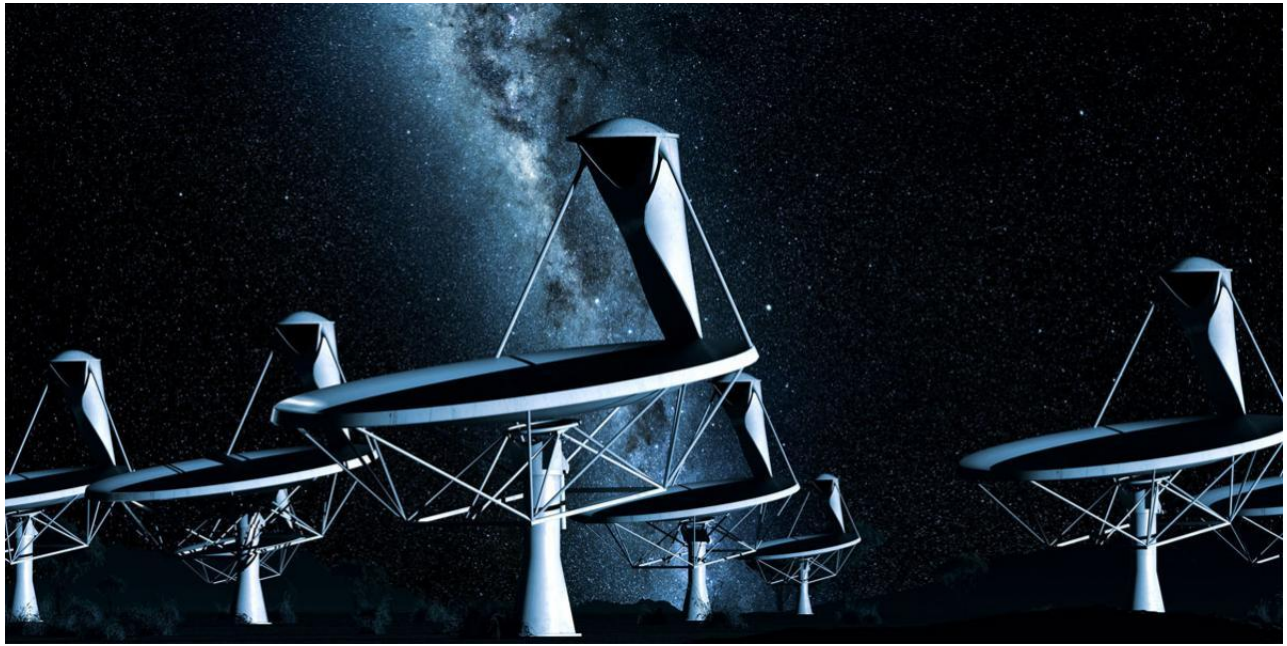
LMC Simulation Framework



*This work is based on the research supported by the **N**ational **R**esearch **F**oundation (through a flagship project of the SA-INDO collaboration). Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard*

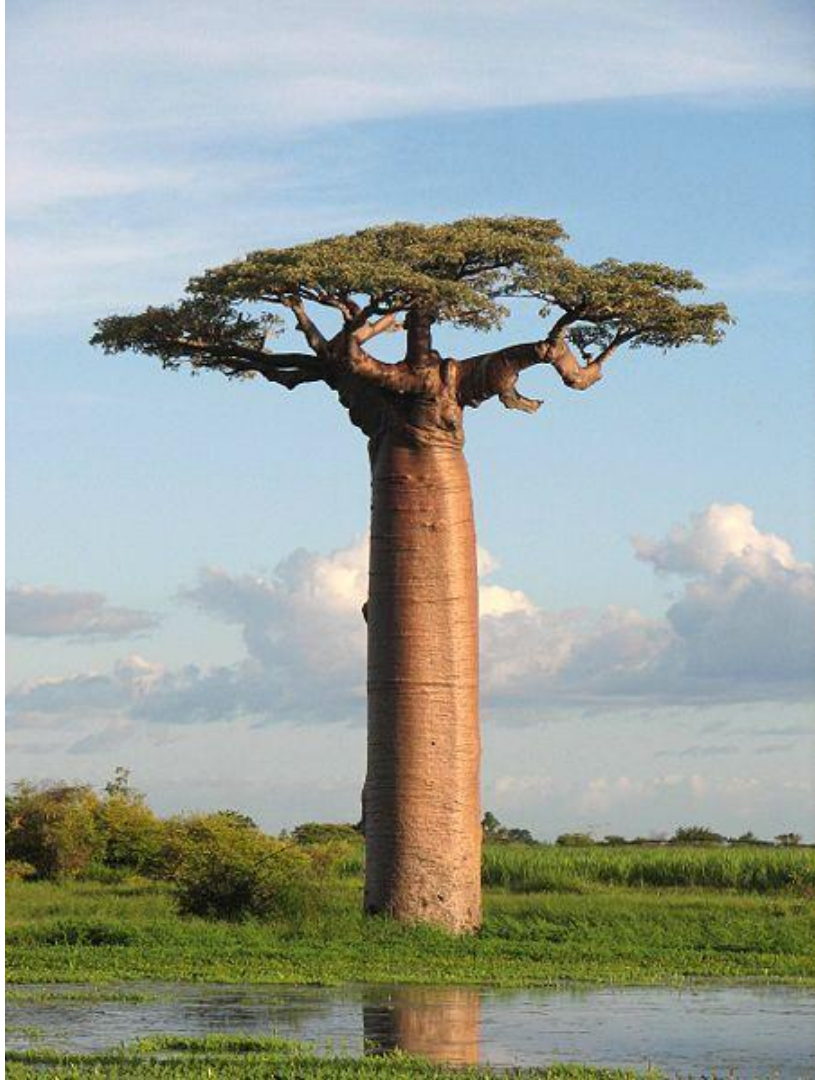
Athanaseus Ramaila
Lize van den Heever
Katleho Madisa
Neilen Marais

Outline

- Iteration 1
  - *MeerKAT CAM (M+ TM) using Tango simulators*
- Iteration 2
  - *Evaluate TANGO tool capabilities in the context of a MeerKAT-like radio telescope*
- Iteration 3
  - *Improved Data-driven Simulation framework, including behaviour extension*
- Iteration 4
  - *Exploratory Simulation of LMC Elements using Simulation Framework, Integration with DSEE*

ITERATION 1

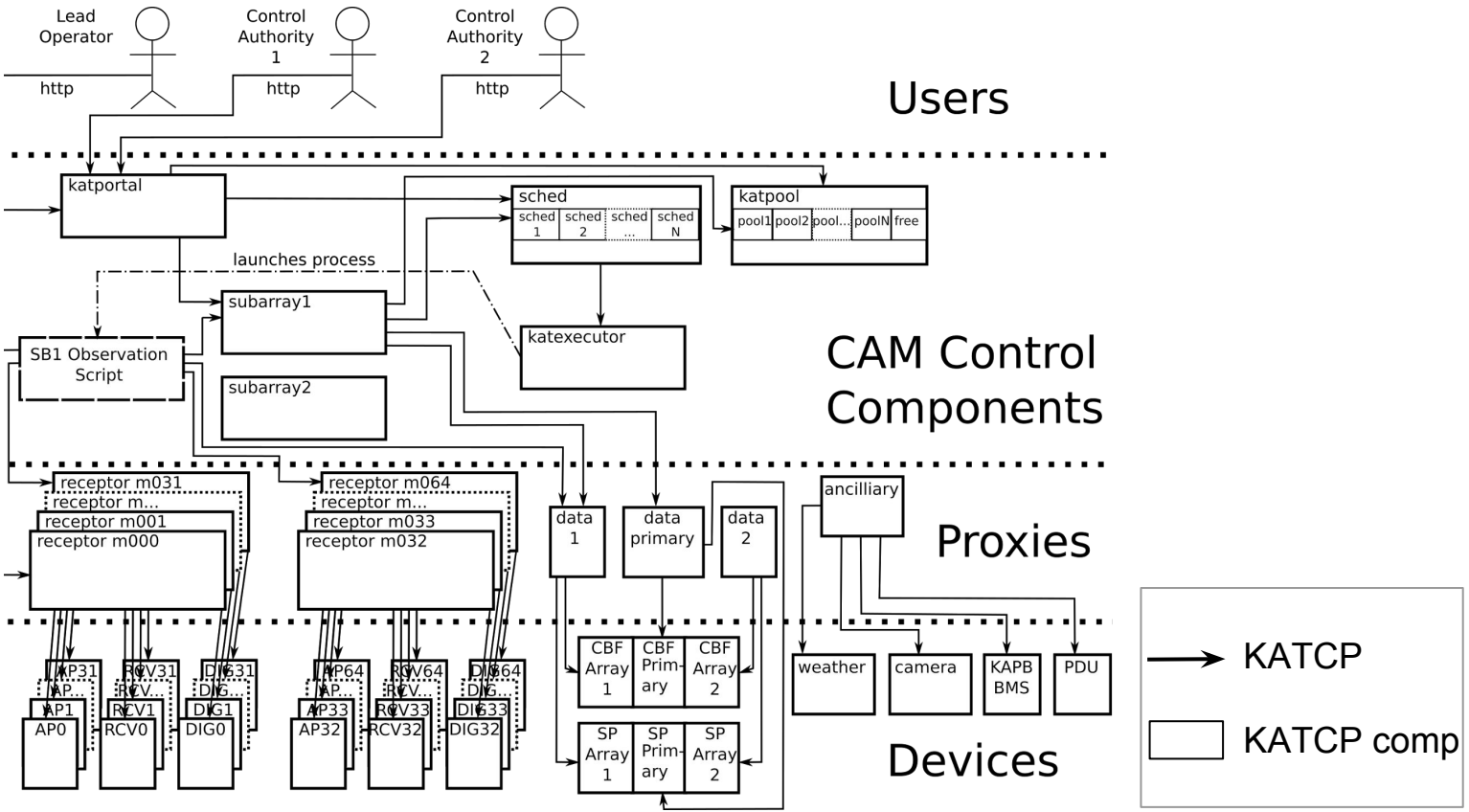
MeerKAT CAM (M+ TM) using Tango simulators



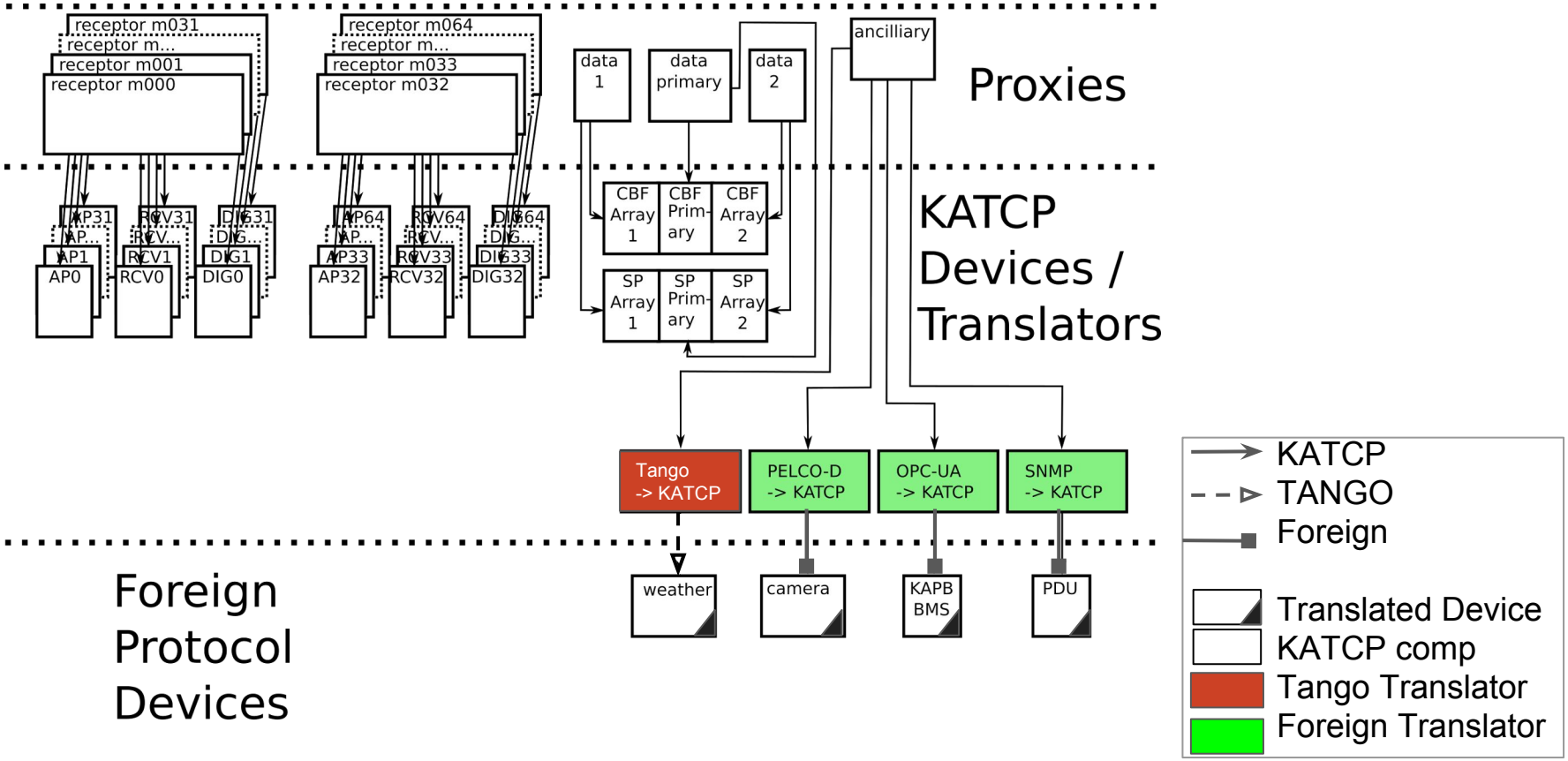
Why TANGO with MeerKAT?

- It's the future (of SKA)
- MeerKAT's future is SKA
- Learn about TANGO by integrating with MeerKAT
- SKA TM Precursor

MeerKAT Architecture



What difference does a TANGO make?

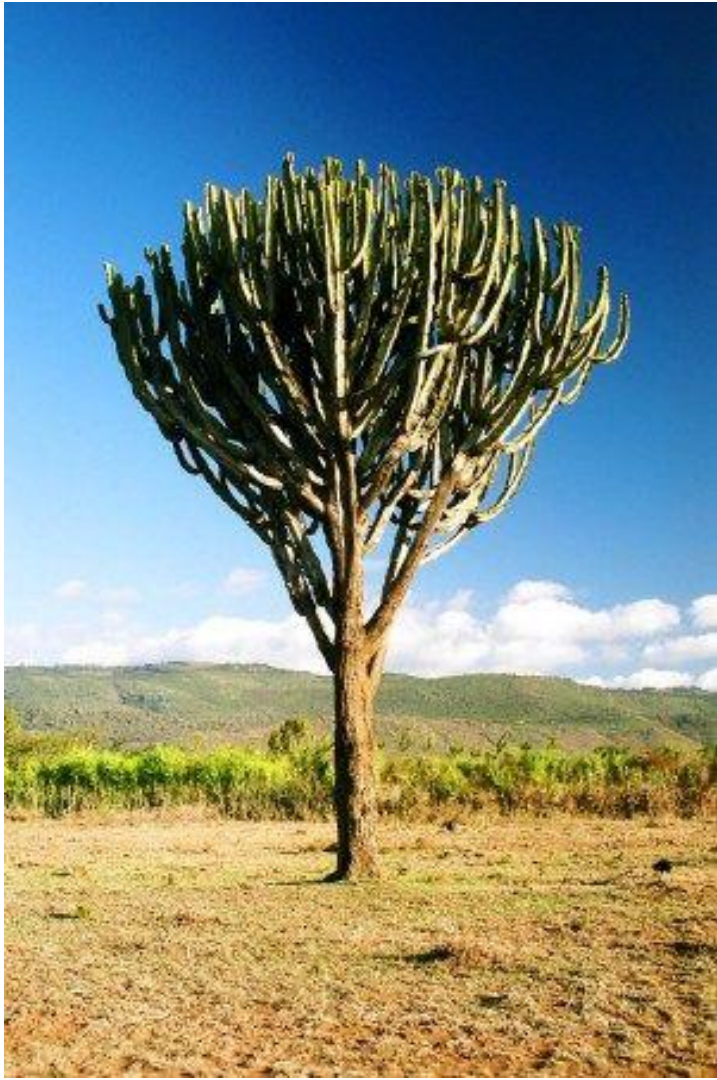


Iteration 1 Learnings

- Generic TANGO -> KATCP translator works well
  - KATCP "device model" mostly superset of TANGO "device model"
- Easy to integrate TANGO devices in existing MeerKAT
- Allows for early SKA subsystems testing with MeerKAT

ITERATION 2

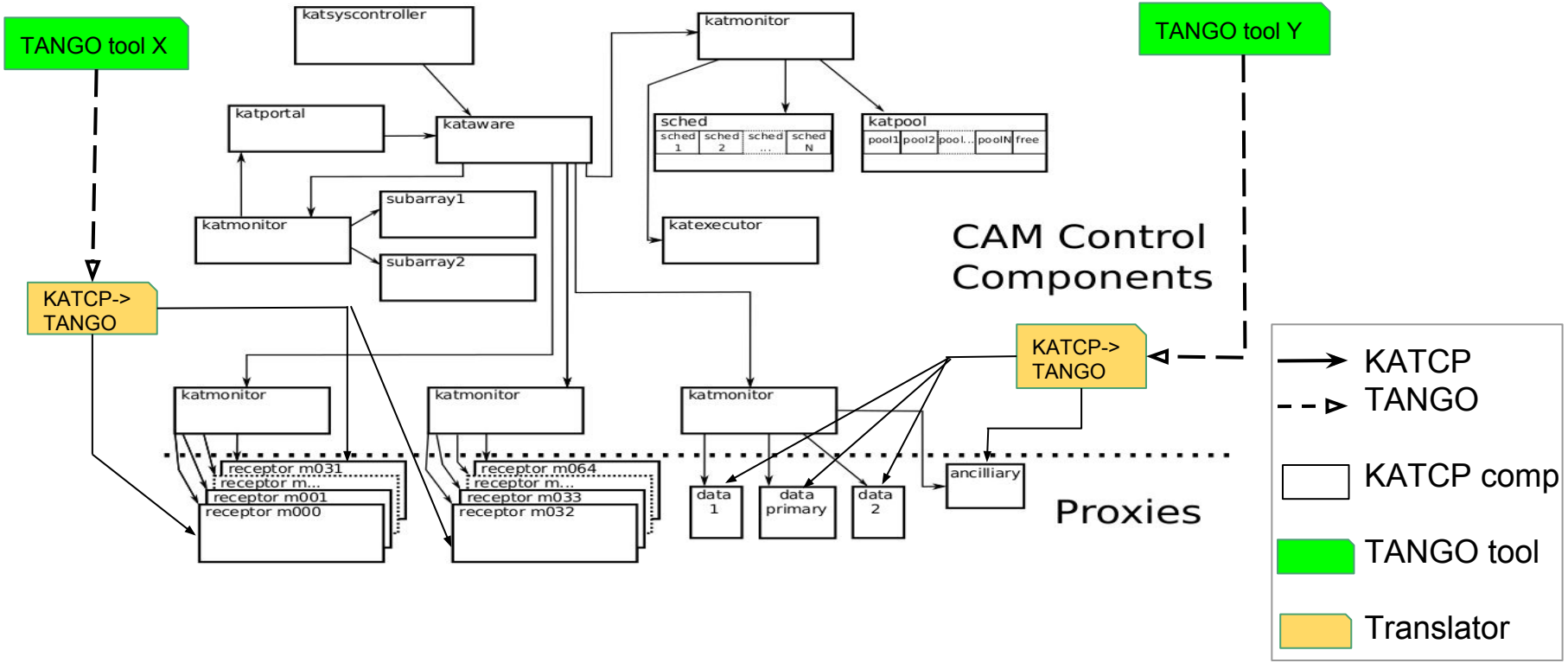
Evaluate TANGO tool capabilities
in the context of a MeerKAT-like radio telescope



The TANGO tools that we evaluated...

- The PANIC Alarm System
  - Report
- The Historical DataBase++ Archiving System
  - Report
- TangoDB Configuration Database
  - Report
- iTANGO CLI
  - Report

Actual MeerKAT architecture (with the TANGO tools)



Iteration 2 Learnings

- TANGO community ecosystem provides useful tools
  - No compelling reason to replace MeerKAT components at present
- Generic KATCP -> TANGO translator works OK
  - Some KATCP "device model" features hard to represent in TANGO
  - Could potentially be addressed (pipes?)
- Potential use for SKA <-> KATCP interop

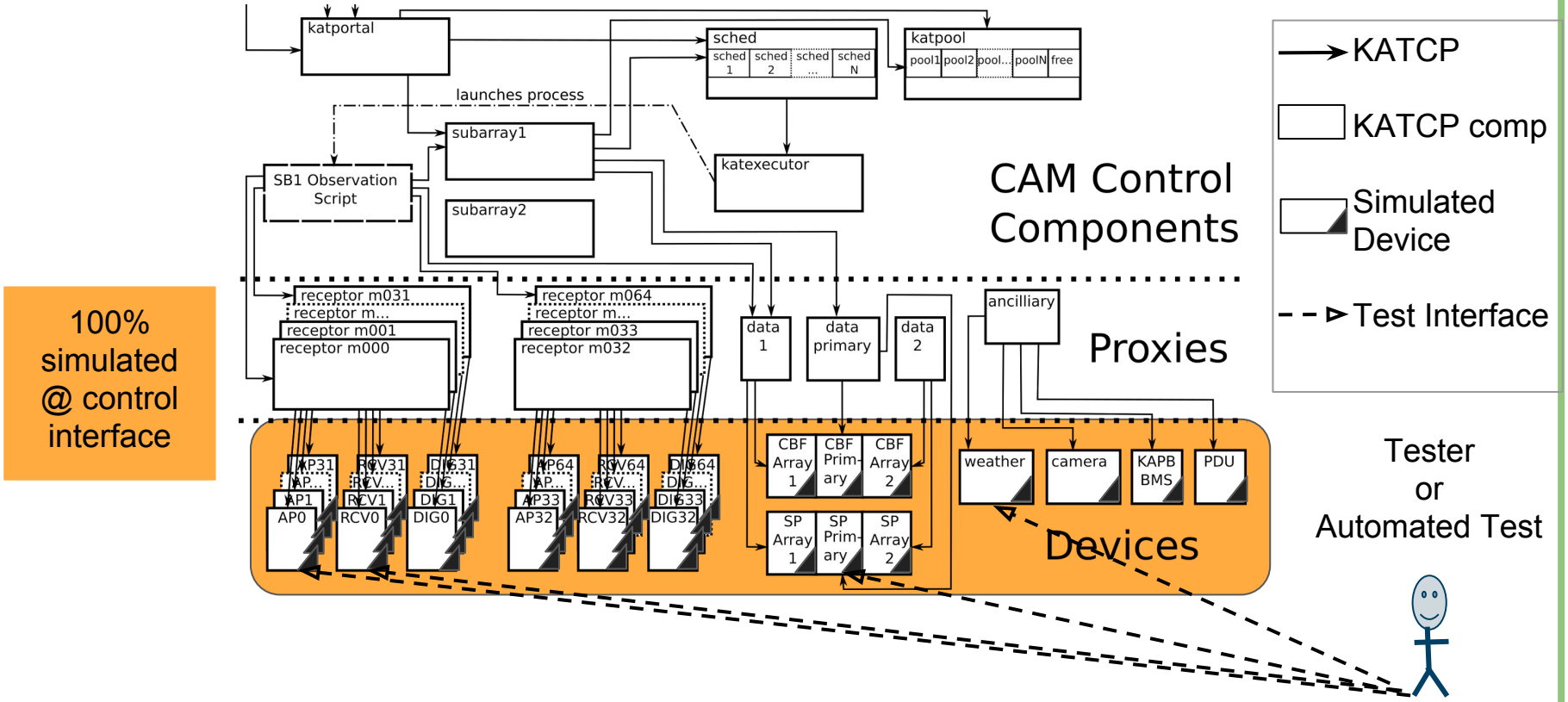
ITERATION 3

Improved Data-driven Simulator framework, including behaviour extension

Why Simulated Devices

- Positive MeerKAT / KAT-7 experience
- To be used by the Element Consortia to develop LMC simulators.
- Support early development work of Element LMCs
- Also used in the TM test environment
- Enable TM to support early Assembly, Integration and Verification efforts
- Easily configure fully simulated development environments
  - TM Development
  - Automated functional/ Integration testing
  - Lab integration with partial simulation

Development / simulated MeerKAT Architecture



How the Simulator is launched

Step 0

```
$ pip install tango-simlib
```

Step 1

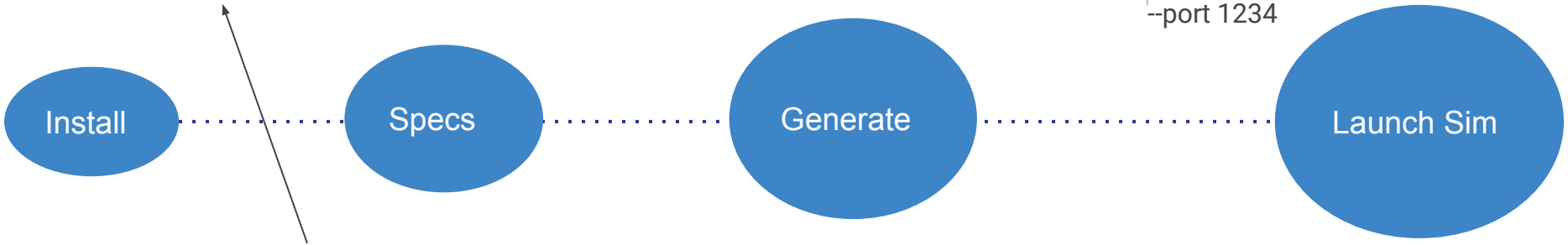
Basic Sim Spec (POGO)
Complex Sim Spec (SimDD)

Step 2

```
$ tango-simlib-tango-simulator-generator \
--sim-data-file Dish.xmi \
--sim-data-file Dish_SIMDD.json \
--device-name DishElement-DS \
--directory /usr/local/bin/
```

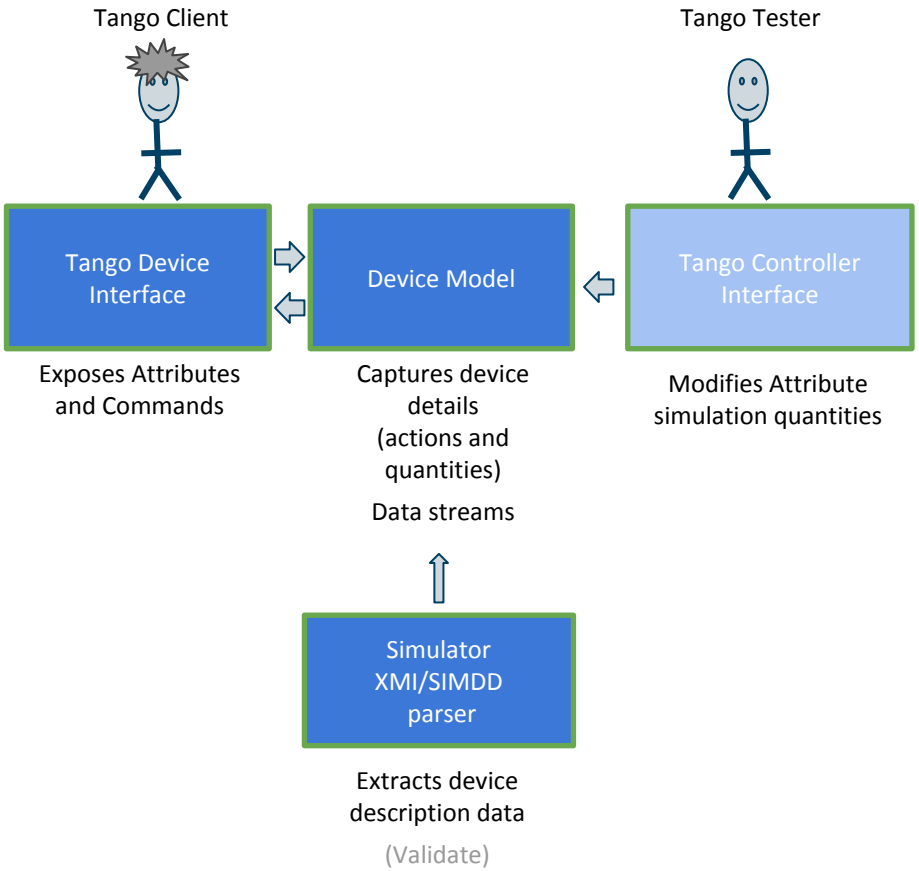
Step 3

```
# If already  registered in tango db
$ DishElement-DS  dish-000

# auto-register
$ tango-simlib-tango-launcher
--name DSH/element/ap000 \
--class DishElement \            --name
DSH/element_ctrl/ap000 \ --class
DishElementControl \
--server-command  DishElement-DS\
--server-instance dish-000 \
--port 1234
```



https://github.com/ska-sa/tango-simlib

Simulator with a Controller Interface



Generated Dish executable script

Note: there is no code generation, just an executable script. The code behaviour is "dynamically" introduced from the Dish.xml and Dish_SIMDD.json specification files.

```
$ tango-simlib-tango-simulator-generator  --sim-data-file Dish.xmi --sim-data-file Dish_SIMDD.json  --dserver-name DishElement
$ cat DishElement.py
```

```python
#!/usr/bin/env python
from PyTango.server import server_run
from tango_simlib.tango_sim_generator import (configure_device_model, get_tango_device_server)


def main():
    sim_data_files = ['/.../Dish.xmi',
                      '/.../Dish_SIMDD.json']  # From command line --sim-data-file(s)
    model = configure_device_model(sim_data_files)
    TangoDeviceServers = get_tango_device_server(model, sim_data_files)
    server_run(TangoDeviceServers)

if __name__ == "__main__":
    main()
```

Basic Simulators

- Uses only the Tango POGO interface generation tool (XMI file)
- Attributes are mapped to model quantities without writing any code
- Commands are mapped to default no-op model actions
- Simulation control interface included, used to manipulate the simulator and induce conditions/failures

Complex Simulators

- Uses simulated device description format to describe simulator behaviour (SimDD JSON file)
- Specify attribute parameters and quantity simulation types
- Override or Modify default command actions using the SimDD
- Custom action handler overrides can be coded in Python
- Simulation control interface, as above

Simulation Parameters

```
"dynamicAttributes": [
        {
            "basicAttributeData": {
                "name": "an_attribute",
                // Other “standard” attribute parameters that can also
                // be specified in the XMI file has been removed for brevity
                "dataSimulationParameters": {
              "gaussianSlewLimited": {
                 "min_bound": "minimum bound for randomly varied simulator quantity",
                 "max_bound": "maximum bound for randomly varied simulator quantity",
                 "mean": "mean value",
                 "max_slew_rate": "maximum slew rate",
                 "update_period": "update period"
                    }
                },
    ]
```

Basic attribute simulation categories:

- *guassianSlewLimited - min/max bounds, mean value, slew_rate and update_period*
- *constantQuantity - initial_value, attribute_quality*

Basic command simulation categories:

- *Input parameter transform - Take an input parameter, applies a transform and place output in a temporary variable*
- *Side effect - Simple action that can modify a simulation quantity or internal state variable*
- *Output return - Return value or exception*

Complex Simulators

To simulate more complex behaviour the commands can be overridden by implementing and specifying an Override Class.

This allows for full flexibility as the complete simulation model can be replaced, if required.

Override actions in the override class are prefixed with action then the name of the command on the TANGO device.

Simulator Data-Description file

```
"override_class": {
    "name": "unique_override_identifier",
    // "module_directory": "Locate the override module in this directory [optional]",
    "module_name": "tango_simlib.examples.override_class"
    "class_name": "OverrideDish"
}
```

Override class

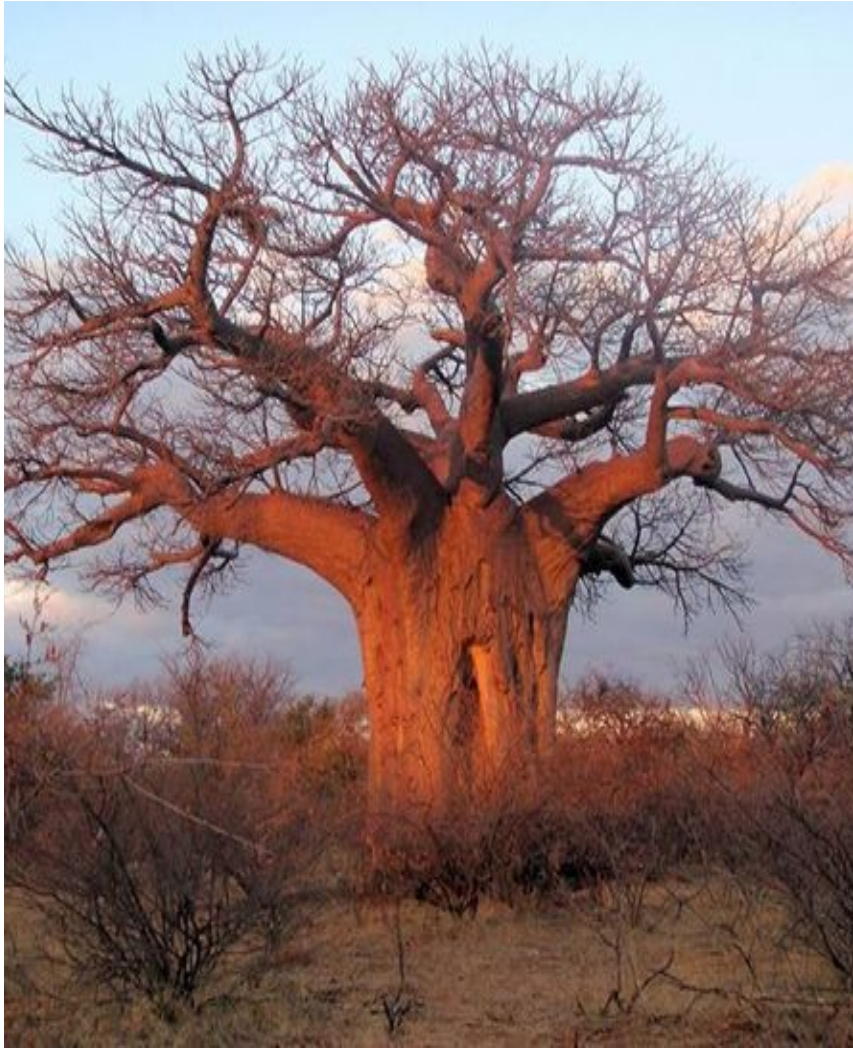
```
class OverrideDish(object):
    """An example of the override class for the TANGO device class 'SkaDishMaster'.
    It provides implementations of the command handler functions for the commands
    specified in the POGO generated XMI data description file.
    """
    def action_slew(self, model, tango_dev=None, data_input=None):
        """The Dish is tracking the commanded pointing positions within the specified
        TRACK pointing accuracy.

        data_input: list
           [Timestamp]
           [azimuth]
           [elevation]
        """
        _allowed_modes = ('OPERATE')
        :
```

Iteration 3 Learnings

- Ported MeerKAT simulator+test interface model to TANGO
  - Released as FOSS : https://github.com/ska-sa/tango-simlib
- Simple simulators are easy to generate
  - MeerKAT experience: covers 80% of use cases
- Complex simulators can leverage base functionality

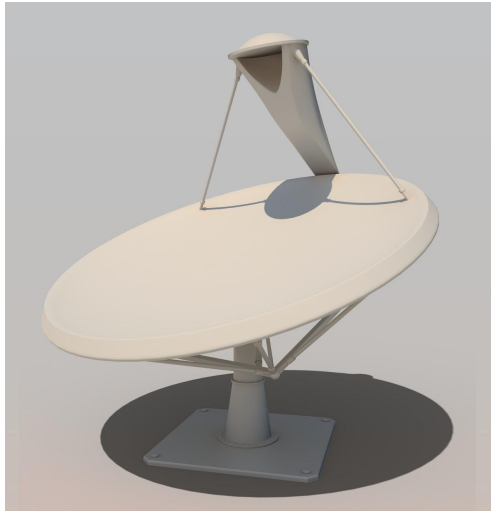
ITERATION 4

Exploratory Simulation of LMC Elements using Simulation Framework, Integration with DSEE

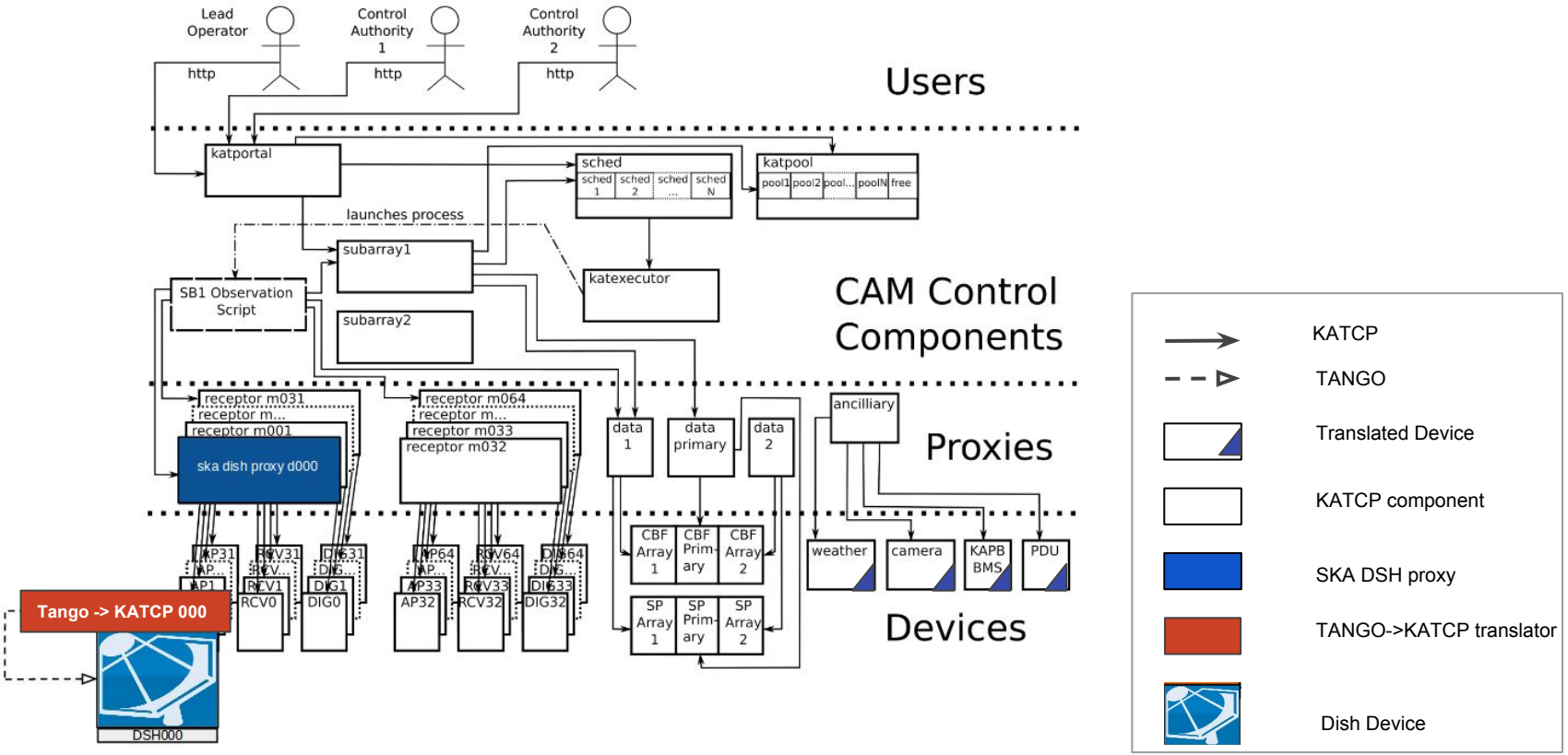


Integrate SKA DSH Element simulator in MeerKAT

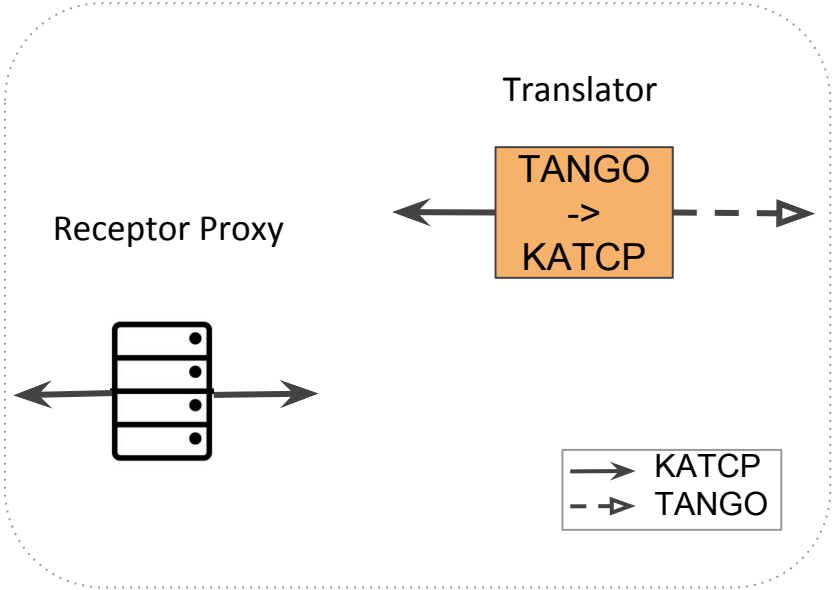
- Will be first (real) element to be integrated
- Prototype to be qualified with MeerKAT
- Preliminary DSH ICD available
- Ported the MeerKAT AP physical model to tango-simlib based DSH simulator
- Integrate DSH simulator with MeerKAT using translator
    - Get MeerKAT ready for DSH prototype!



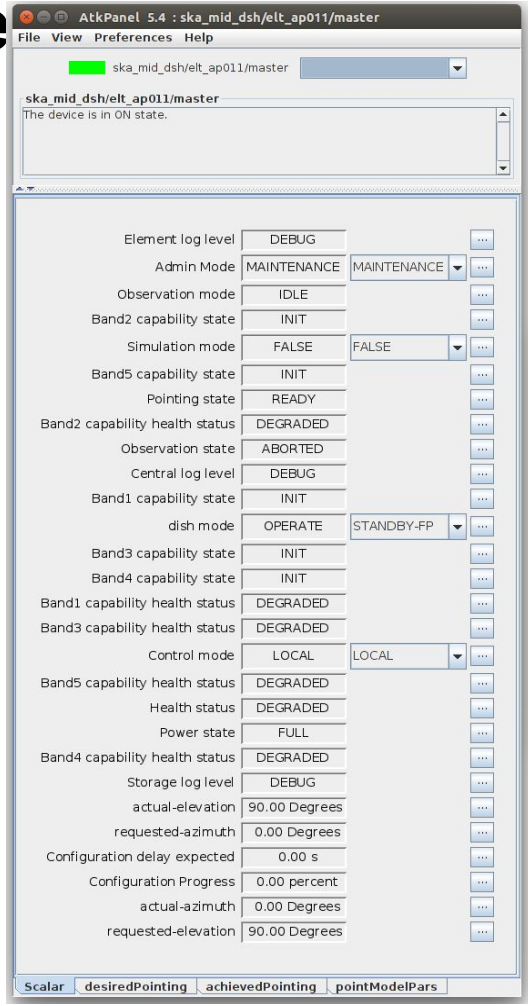
MeerKAT-TANGO system



Connecting To Dish Simulator Device

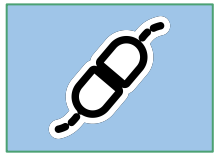


Dish Simulator



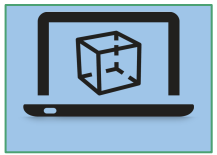
SKA DISH Proxy

- Provides standardised MeerKAT/KAT-7 high-level interface
  - Development based on existing MeerKAT receptor proxy



System component/tools (TM) connect via the receptor proxy (LMC) not directly to DISH

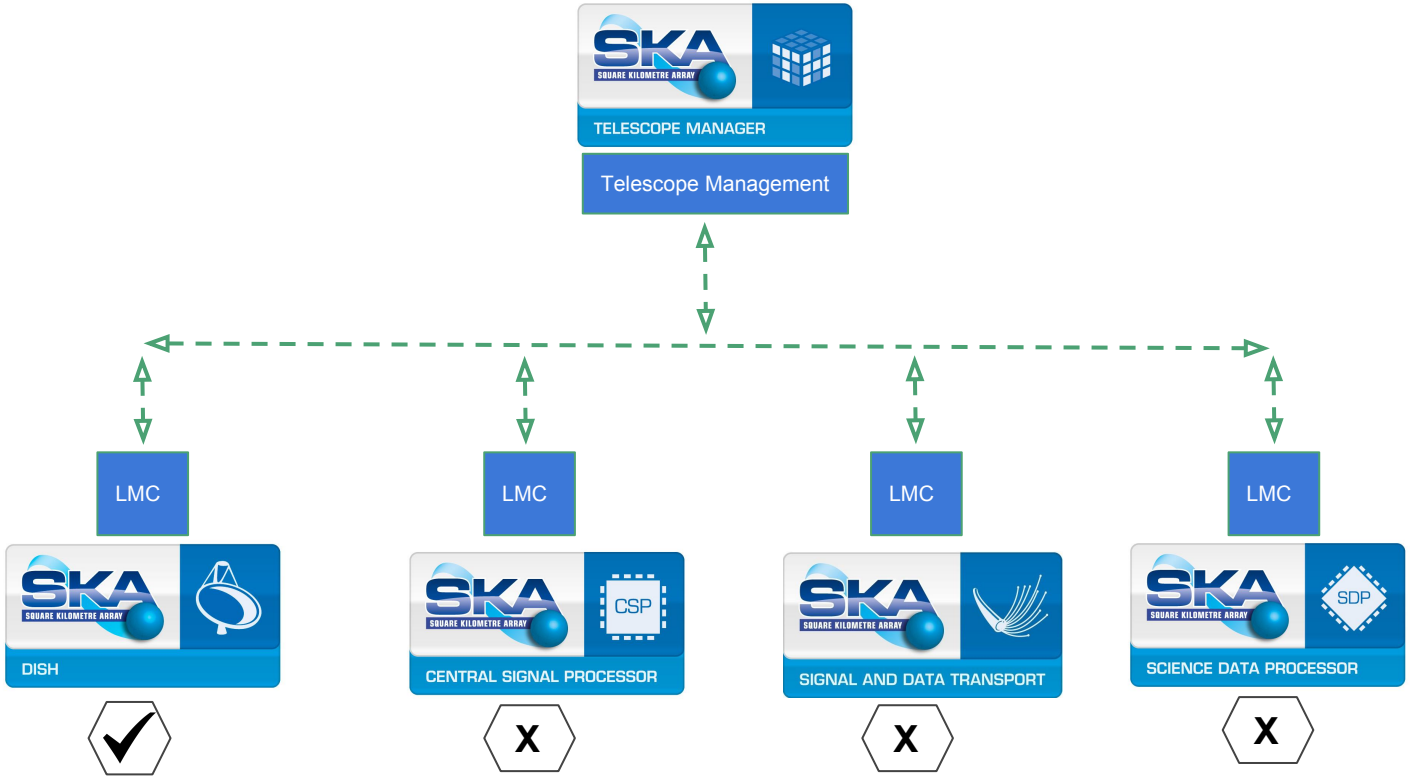
Allows the rest of the MeerKAT CAM system to interface with the DISH device.



The proxy is responsible for managing devices and exposing their KATCP interface.

Allows simultaneous observation with MeerKAT receptors and prototype DISH

Simulated SKA (MID) Telescope



Data-driven simulator tools

- **DSEE**: Generates Simulator Description files (SimDD)
  - https://gitlab.com/patwari.puneet.ska/MAC-SEEN
- **tango-simlib**: Simulator interface as per SimDD and Controller interface to manipulate the simulator
  - https://github.com/ska-sa/tango-simlib
- **mkat-tango**: TANGO/KATCP Device Translators

DEMO

weather_SIMDD.json

```
{
    "dynamicAttribut
        {
            "basicAt
                "nam
```

AtkPanel 5.4 : mkat_sim_control/weather/1

File View Preferences Help

mkat_sim_control/weath...

std_dev	2.00	0.00
mean	40.00	40.00

Scalar

```
},
```

Alarms

1:17:30 LST 08:14:56 UTC 10:14:56 local

cam@ska.ac.za (Lead Operator)

LO: cam@ska.ac.za

4 1

Current Alarms

Search Alarms... KNOW ACKNOWLEDGE CLEAR Show Cleared Alarms

Severity	Date Changed	Priority	Name	Message
	16-05-2017 08:14:03	new	ANC_Wind_Gust	critical,new,anc_gust_wind_speed value = 19.8432 >= 16.9
	16-05-2017 00:24:17	new	PSQL_Disk_Space_Used	warn,new,anc_dbmon_varlibpostgresql_disk_space value = 70 >= 70
	25-04-2017 22:08:04	new	monctl_Disk_Use	warn,new,anc_ganglia_monctl_kat_disk_part_max_used value = 75.0005 >= 75
	25-04-2017 08:52:03	new	Varkatconfig_not_up_to_date	warn,new,agg_anc_cam_varkatconfig_uptodate is False
	25-04-2017 08:52:02	new	Camera_RFI_Risk	warn,new,agg_anc_camera_rfi_risk is True
	16-05-2017 08:12:11	new	No_Lead_Operator	nominal,new,katpool_lo_id value = 'cam@ska.ac.za'. status = nominal.
	25-04-2017 10:02:01	new	ANC_Wind_Speed	nominal,new,anc_mean_wind_speed value = 10.5510314102886 is in NOMIN...
	25-04-2017 10:00:05	new	System_KAPB_Temperature_Failure	nominal,new,agg_anc_kapb_temperature_ok value = True. status = nominal.
	25-04-2017 10:00:05	new	ANC_Wind_Reporting_Failure	nominal,new,agg_anc_wind_reporting_ok value = True. status = nominal.

File View

mkat_si

The devic

Alarm : Va

Ou

Ba

Input communication OK

Scalar

Thank you!