MEERKAT POSTER AND DEMO CONTROL AND MONITORING HIGHLIGHTS

C. de Villiers, SKA, South Africa

Abstract

MeerKAT is a 64-dish radio telescope that is being built in the arid Karoo region of South Africa's Northern Cape Province. On its completion in 2016, MeerKAT will be the most powerful radio telescope in the Southern Hemisphere, and will hold this title until the commissioning of the much larger Square Kilometre Array (SKA) in about 2024. The SKA project is to be split between Southern Africa, Australia and New Zealand, with the core of the Southern African array to be built at the Karoo site.

MeerKAT has a precursor in the 7-dish KAT-7 engineering prototype array, which is already in operation at the same site. KAT-7 has been commissioned and is producing valuable scientific results. KAT-7 is a technology demonstrator that has helped to generate and test many of the design ideas that will now find their way into MeerKAT and perhaps ultimately into SKA as well.

This poster summarizes some of the features of the Control and Monitoring (CAM) architecture for MeerKAT [1].

INTRODUCTION AND BACKGROUND

The MeerKAT telescope will consist of sixty-four 13.5 metre dishes with an offset Gregorian configuration. This is also the reference dish design for the mid-band section of the SKA . The dish baselines will range from 29 m to 20 km, and the telescope will observe in three frequency bands:

- 0.58 – 1.015 GHz
- 1 – 1.75 GHz
- 8 – 14.5 GHz

PROPOSED MEERKAT APPLICATIONS

Construction is just beginning on MeerKAT, but it is already in demand for a number of planned research projects. These include:

- Investigating the physics of neutron stars through pulsar observations
- An ultra-deep survey of neutral hydrogen gas in the early universe
- A search for CO at high red-shift ($z > 7$) to investigate the role of molecular hydrogen in the early universe.
- A survey for atomic hydrogen and OH lines in absorption against distant continuum sources
- Investigations of different types of galaxies, dark matter and the cosmic web
- Searching for and investigating new and exotic pulsars
- Galaxy formation and evolution in the cluster environment
- Galactic structure and dynamics, distribution of ionised gas, recombination lines, interstellar molecular gas and masers
- Deep continuum observations of the earliest radio galaxies
- Studies of gamma-ray bursts, novae and supernovae, plus new types of transient radio sources
- Participation in global VLBI operations with all major radio astronomy observatories around the world. MeerKAT will add considerably to the sensitivity of the global VLBI network.
- Further potential science objectives for MeerKAT are to participate in the search for extraterrestrial intelligence and collaborate with NASA on downloading information from space probes.

CONTROL AND MONITORING OVERVIEW

The Control and Monitoring (CAM) department of SKA South Africa is responsible for the collection and archiving of sensor data, the control of dishes and correlators, and the provision of a Graphical User Interface (GUI) for operator interaction with the telescope. On the detection of abnormal conditions, CAM must generate alarms which may trigger automated safety measures or simply alert the operators.

All the control and monitoring software is written in Python. In addition to the GUI there is an IPython-based interactive command-line and scripting interface that is useful both for developers and for sophisticated users.

KATCP Protocol

KATCP is the standard protocol used for all CAM communications. It is a simple text-based protocol with a TCP transport layer and a few higher-level constructs that make it more useful for control and monitoring purposes. Among these is the concept of a KATCP Sensor, a software-defined monitoring point that may or may not represent a real hardware device. A Sensor can push updates to multiple subscribers, each of which may set its own subscription strategy that determines how often, and in what conditions it receives values and status from the Sensor. Standard strategies include:

- Periodic
- Event-based

ISBN 978-3-95450-139-7

- Event-based with guaranteed minimum and maximum update intervals.

Sensors present a standard and unifying interface for both analog and digital values. Every sensor value update is accompanied by an indication of sensor status and two, millisecond-precision timestamps – one for the latest change of the sensor's value, the other for the time it communicates this value to its subscriber. This standard interface makes it easy to define rules for sensor aggregation and averaging as well as rules for alarms.

KATcp's request-response protocol implements standardised monitoring schemes such as watchdogs and heartbeat queries. In addition a 'help' command allows automated introspection of any KATcp server, and hence dynamic configuration across a cluster. A new server can be automatically queried for its available sensors and commands, which can then be exposed with no programming input via system displays, menus and the command line. IPython's auto-completion feature provides an especially productive mechanism for interactive exploration of the system.

CAM's core libraries (katcorelib, katuilib) build on the KATcp concepts to create a single object-oriented access hiererchy for the entire telescope configuration. Components are made available as attributes of the parent container object, while sensors and commands are exposed as attributes of the components.

Proxy Architecture

One of the most powerful concepts in CAM is its **proxy architecture**. This concept is based on the consistent insertion of a KATcp server layer in front of every hardware or software component and device. Every other part of the system then accesses the proxy, not the 'real thing'. This brings the immediate benefit that every system component has a predictable and introspectable interface. The proxy can add intelligence to an interface, as for example the receptor proxy does by implementing a sophisticated ephemeris-based tracking algorithm, supplemented by multiple pointing corrections, that hides the antenna's raw azimuth and elevation controls. The proxy layer also allows a layer of protection for hardware, as the proxy can filter undesirable commands and parameter values before they reach the device.

The proxy concept also lends itself to the deployment of a fully-simulated system. Since the only hardware dependencies are in the proxies themselves, a component simulator need only implement the same KATcp protocol as the real device. Proxies also make it easy to provide access control for resources.

Configuration Management

KATconfig is the component supporting static configuration. It is implemented using a Python templating library, and thus readily supports configuration of multiple instances of any component via parameter substitutions. Configuration is read from a single, version-controlled central location, then made available to all nodes and servers via an XMLRPC server. Among the configuration items handled by KATconfig are:

- Assignment of processes to nodes
- Sequence of process startup
- Process command-lines and parameters
- Process logging handlers, levels and destinations

Observation Control

The **Observation Control Framework** has been developed to support the MeerKAT 'subarray' concept – allowing multiple concurrent independent observations, each using a subset of the telescope resources.

- **katsyscontroller** manages the ordered startup and shutdown of servers and components and also provides facilities for emergency stopping and stowing of the dishes, and protective shutdown of the system.
- **katscheduler** allows for automated and manual scheduling of observations.
- **katpool** allocates system resources.
- **katexecutor** manages script execution during an observation and handles output and logs.

CONCLUSION

The poster highlights just a few of the major architectural components of MeerKAT. Based on experience gained with KAT-7 and the enhancements that have been made to that design,, we believe this architecture has the flexibility to cope with the technical challenges posed by the much larger telescope.

REFERENCES

[1] L. van den Heever and N. Marais, "MeerKAT CAM Design Description Rev. A" (2013)

ISBN 978-3-95450-139-7