ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-TUPDP069

AVN RADIO TELESCOPE CONVERSION SOFTWARE SYSTEMS

R.L. Schwartz*, P.J. Pretorius†, R.M. Ebrahim§,
South African Radio Astronomy Observatory (SARAO),
Cape Town, South Africa

Abstract

The African VLBI Network (AVN) is a proposed network of Radio Telescopes involving 8 partner countries across the African continent. The AVN project aims to convert redundant satellite data communications ground stations, where viable, to Radio Telescopes. One of the main objectives of AVN is human capital development in Science, Engineering, Technology and Mathematics (STEM) with regards to radio astronomy in SKA (Square Kilometer Array) African Partner countries.

This paper will outline the software systems used for control and monitoring of a single radio telescope. The control and monitoring software consists of the User Interface, Antenna Control System, Receiver Control System and monitoring of all proprietary and Off-The-Shelf (OTS) components. All proprietary and OTS interfaces are converted to the open protocol (KATCP).

INTRODUCTION

The African Very Long Baseline Interferometry Network (AVN) is a pan-African project that will develop Very Long Baseline Interferometry (VLBI) observing capability in several countries across the African continent, either by conversion of existing telecommunications antennas into radio telescopes or by building new ones.

This paper focuses on the conversion of the Kuntunse satellite communication station (near Accra, Ghana), specifically the software systems developed by the software groups of SARAO.

BACKGROUND

African VLBI Network

The AVN is an ambitious project that will establish a network of radio telescopes across the African continent to support the existing international VLBI network. Stations located across Africa would greatly improve the image fidelity of VLBI observations, since the geographic location of these stations improves sensitivity to angular scales on the sky that are not well sampled using the currently available global configuration [1].

Objective of the AVN

The AVN will help to develop the skills, regulations and institutional capacity required in SKA partner countries to optimise African participation in the SKA and enable participation in SKA pathfinder technology development and science. The SKA AVN partners of South Africa are: Botswana, Ghana, Kenya, Madagascar, Mauritius, Mozambique, Namibia, and Zambia.

The AVN programme will transfer skills and knowledge to African partner countries to build, maintain, operate and use radio telescopes.

It will bring new science opportunities to Africa on a relatively short time scale and develop radio astronomy science communities in the SKA partner countries.

VLBI

VLBI is astronomical interferometry used in radio astronomy, where the same astronomical source signal is recorded by multiple radio telescopes on Earth or in space. The data is timestamped with very accurate time sources (e.g., GPS and hydrogen maser). These signals are then correlated at a later stage to produce the resulting image, using the timestamps. This emulates a telescope with a diameter equal to the maximum distance between the telescopes and enables distances between antennas to be much greater than possible with conventional interferometry.

STATION CONTROL AND MONITORING SOFTWARE OVERVIEW

The Software Group

The AVN Software Group currently consists of three members who are responsible for the main Control and Monitoring software systems design, implementation and commissioning, as well as the interconnection and networking of various subsystems on the telescope.

KATCP

KATCP (KAT Communications Protocol) is a simple ASCII communications protocol layered on top of TCP/IP. It was developed as part of the Karoo Array Telescope (KAT), and is used for the monitoring and control of hardware devices.

The protocol consists of newline separated text messages sent asynchronously over a TCP/IP link. There are three categories of messages: requests, replies and informs. The AVN control and monitoring system uses KATCP as the primary station communications protocol.

Station CAM Software

AVN Station Control and Monitoring (CAM) Software consists of the following software subsystems:

- Station Controller software
- Protocol Translation software
- Antenna Steering Control System software
- Environmental Monitoring System software
- VLBI Backend software

*rschwartz@sarao.ac.za
† ppretorius@sarao.ac.za
§ rebrahim@sarao.ac.za

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-TUPDP069

There are other software subsystems developed by SARAO that form part of the radio telescope, but are not included in the Software Group's responsibility. These includes the Receiver software, Maser Building Control and Monitoring software, and the Single Dish backend.

The Station Controller software monitors all the subsystems (hardware and software) that form part of the radio telescope. In addition to monitoring, the Station Controller software controls the Antenna Steering Control System (ASCS), the receiver, and the Single Dish Backend (SDB). It also provides an interface to the VLBI backend via Field System (FS) for control. All control and monitoring is done via the KATCP protocol. The tracking algorithms of the radio telescope are also implemented on the Station Controller software.

The Protocol Translations software interfaces to several other hardware and software subsystems and translates their sensors into KATCP sensors.

The ASCS software controls and monitors the antenna subsystem, including the servo drives and all the digital and analogue input and outputs of the antenna.

The Environmental Monitoring System (EMS) software interfaces with the environmental sensors, which include sensors for: wind direction, wind speed, absolute pressure, relative humidity, and temperature.

The VLBI Backend software is Field System, by NASA [2], and provides station software that is customised for this radio telescope.

STATION CONTROLLER SOFTWARE

Overview

The Station Controller Software (SCS) consists of multiple components working together to control and monitor the radio telescope. Some of the functions of the SCS are as follows:

- Collects sensor data and makes it available for visualisation and logging to a database.
- Provides a KATCP server interface with all the control and monitoring for all subsystems.
- Provides control of radio telescope components via KATCP (e.g., ASCS, Receiver, Backends).
- Provides different levels of user control.
- Provides the ability to generate and extract reports from the database.

The SCS integrates with a number of subsystems:

- Antenna Steering Controller System
- Receiver System
- Environmental Monitoring System
- Infrastructure Controller
- Digital Backends

The User controls and monitors the SCS. All subsystems provide interfaces for control and/or monitoring of hardware sub-components as shown in Figure 1.

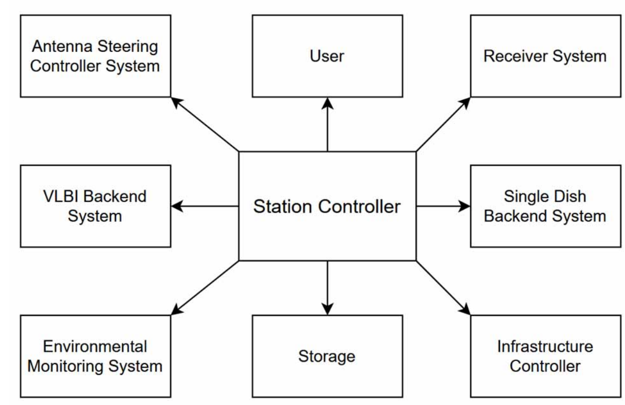

Figure 1: SCS Interfaces.

Following Systems Engineering principles, requirements were gathered from the User Requirement Specification for the Telescope, and a Software Requirement Specification was developed.

Design and Architecture

The main goal of the station controller was to provide a graphical interface to monitor and control the telescope subsystems in an intuitive manner. An investigation was done to determine if it was feasible to reuse the current MeerKAT CAM system, but this proved to be too technical and had too much overhead for this single dish telescope, and it was decided to design and create a system more suited to the project.

With the skill-set of the AVN Software Team, being C, C++ and a bit of C#, a rudimentary investigation was done to select the programming language that the SCS would be developed. The Qt framework (based on C++) was widely used in the control and monitoring industry at the time, but with the more modern and more widely used in the holistic software industry, it was decided to develop the application in C#, thereby making it easier to find software developers if required.

The architecture was designed to follow a functional approach, broken down into the following functions:

- Communications - communication between the subsystems via KATCP.
- Monitoring - monitoring of the overall health, status and fault-detection of the subsystems.
- Logging - storage of subsystem sensor and error data, as well as user reports.
- Control - direct control of aspects of controllable subsystems.
- User - management of roles and accessibility of the functionality through the user interface (UI).

Initially, the software was intended to be run on the Linux Operating System (OS), and developed using the cross-platform framework, Mono. We found the tools had not sufficiently matured at that time, and the UI displays did not present well. We therefore opted to use the Windows OS to host station controller software.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-TUPDP069

Configuration

The Station Controller is configuration heavy to allow for easy changes to the settings without the need for application updates. This is especially necessary for the KATCP monitoring of subsystem sensors, where the sampling strategy parameters can be adjusted if required, and the sensor displays can be tailored to meet the needs of the operator.

User Access

Access to the user interface is controlled by a role-based access layer, each role having an increasing level of access to certain information and control functions. The roles are classified as follows:

- Operator - basic operational control of the telescope subsystems required to run observations.
- Maintainer - additional access to sensor information and subsystem control for fault investigations.
- Engineer - additional abilities for engineering work and configuration.

User Interface

The interface is presented to users in the form of windows per subsystem for control and monitoring of important sensors. Additional sensor windows are made available to Maintainer and Engineer users for subsystem error and fault investigations.

Sensors are updated on the display as per the KATCP protocol sensor sampling strategy, and indicate to the user the name, value and status of each subsystem sensor.

Logging

All sensor data is logged to a MariaDB database for historical investigative purposes, if required. This time-series data is then read out and displayed in graphic chart form per sensor for a specified period using other tools. The data is periodically optimised and archived to maintain the database performance.

Safety

Safety is inherently built into the subsystems where required, and are conveyed to the user on the display and via an audible alarm. The Station Controller implements an automatic stow safety feature, that involves coordination between multiple subsystems.

The SCS monitors the wind speed from the EMS. If certain configurable conditions are met, like consistent high wind speed, the ASCS is commanded to put the telescope into a stow position as fast as possible. This is to prevent possible damage to the antenna structure.

Testing

Continuing Systems Engineering principles, all requirements are tested using an Acceptance Test Procedure (ATP), either via analysis, demonstration, inspection, or formal testing.

The test reports are versioned and stored in the configuration and change management system for future reference and auditing.

PROTOCOL TRANSLATION SOFTWARE

Background

Following the development of an in-house Input-Output (IO) controller, titled the Infrastructure Controller (ISC), it was necessary to develop translation software between the low-level protocol for reading and writing IOs to the telescope protocol of KATCP, and so the Infrastructure Controller Software (ICS) was designed. This server based software was written in the C++ programming to cater for the fast processing of the proprietary ASC-CAM protocol.

As the project progressed, a number of onsite OTS devices also needed to be monitored, and the ICS was then repurposed and extended to be a translation layer for the various protocols, converting them into KATCP for ingest eventually by the Station Controller, being a KATCP Client.

Interfaces

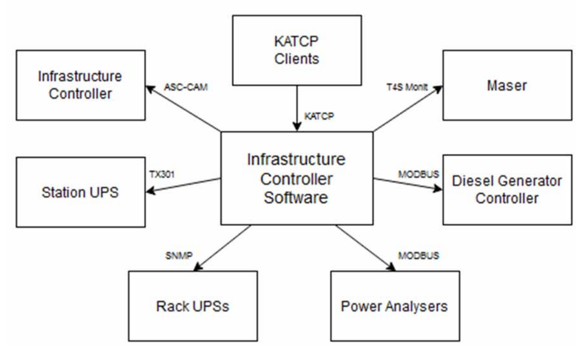

Figure 2: ICS Interfaces.

Figure 2 shows interconnections between the ICS and devices. The ICS currently supports the following protocols, in addition to KATCP:

- ASC-CAM - proprietary protocol used in the Infrastructure Controller device.
- TX301 - Tescom proprietary RS232 based protocol.
- T4S Monit - T4S proprietary protocol over UDP.
- MODBUS - industrial communication protocol.
- SNMP - internet standard protocol for communicating with network devices.

Configuration

The ICS is also heavily reliant on configuration files making it dynamic in operation to allow for setup of the ISC IOs, as well as adding additional SNMP and MODBUS devices easily.

ANTENNA STEERING CONTROL SOFTWARE

Overview

The Antenna Steering Control System (ASCS) software runs on a Programmable Logic Controller (PLC) based system and is used to control the antenna. The software is written in Structured Text (ST) that is part of the IEC 61131 standard for programmable controllers.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-TUPDP069

Interfaces

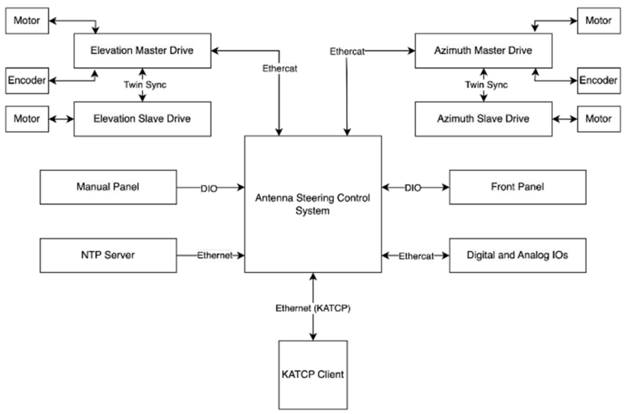

Figure 3: ASCS Interfaces.

Figure 3 shows the ASCS interfaces, which include the:

- Master Servo drives for the elevation and azimuth axis via etherCAT (a real-time Ethernet based network); via the servo drives it also has access to the elevation and azimuth encoders.
- Manual panel that provides manual control over the movement and stow of the antenna via digital inputs.
- The front panel with status indications via digital inputs and a button to stow the antenna via a digital input.
- NTP Server for NTP time synchronisation via Ethernet.
- Digital and analog inputs and outputs via etherCAT using an etherCAT bridge.
- SCS or any other KATCP slave.

The software includes a KATCP server (single client) to:

- Provide an interface to control the antenna, including mode changes, watchdog, setup of Network Time Protocol (NTP) server address and pointing commands.
- Monitor all the sensors of the antenna and provide them as KATCP sensors.

Modes

The following modes are available in the ASCS:

- Idle - parked and braked
- Stow - move to stow position
- Remote point – un-braked and ready to move
- Azimuth reset - reset azimuth axis using a reference point on the azimuth platform

Control

The control loop takes the position commands (time, azimuth and elevation), and feedback from the encoders to calculate the desired speed of each axis. The speed commands are then sent to the servo drives to perform speed to torque control, which is then used to drive the motors. There are two servo drives and motors on each axis, to prevent backlash. Twinsync is used on the servo drives to synchronise the master and slave drives with a torque offset.

Safety

All software safety features are also implemented by the software, including soft limits, speed limits (acceleration is limited by the servo drives). The software will go to idle state (with the motors braked) if there was no watchdog command within 1 second. The antenna will stow automatically if there was no communication for 15 minutes. An alarm will sound when the antenna is about to move. The hardware interlocks (e.g. emergency stops) are also monitored and provided as KATCP sensors.

ENVIRONMENTAL MANAGEMENT SYSTEM SOFTWARE

Overview

The Environmental Monitoring System (EMS) application runs on a Teltonika RUT955 router to monitor environmental sensors. The RUT955 runs an OpenWrt based operating system, Teltonika RUTOS. The software is written in the C++ programming language.

Interfaces

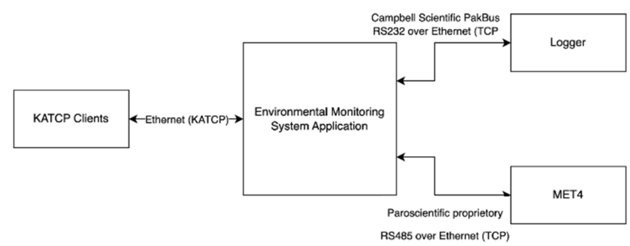

Figure 4: EMS Interfaces.

Figure 4 shows the EMS application interfaces:

- A Logger (Campbell Scientific, Inc CR200X) via Ethernet (TCP). The Logger is connected to the Router via RS232 and the Router converts it to TCP.
- A Meteorological Measurement System (Paroscientific, Inc MET4). The Meteorological Measurement System is connected to the Router via RS485 and the Router converts it to TCP.

Architecture

The software is KATCP server based (multiple clients):

- It monitors battery voltage, wind speed and wind direction with the Logger and provides them as KATCP sensors. Wind speed is used by the SCS to stow the antenna.
- It monitors the absolute pressure, temperature and relative humidity with the MET4, and provides them as KATCP sensors. Absolute pressure, temperature and relative humidity is used by the SCS (and ST in Field System) in the tracking algorithm.

The Logger protocol is Campbell Scientific PakBus and the MET4 protocol is a Paroscientific proprietary protocol. The RUT955 is time synchronised to an NTP Server.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-TUPDP069

FIELD SYSTEM STATION INTEGRATION (VLBI BACKEND INTERFACE)

Field System

The AVN makes use of the Field System software for VLBI Operations. The Field System is a set of interacting programs that form a complete control system for VLBI operations and data acquisition [2]. Field System is Linux based, and is written in the Fortran and C programming languages, with both 32-bit and 64-bit support. The 64-bit, version 10.1 release of the Field System is currently in use in the AVN. Developed by NASA, Field System is extensively used in both space geodesy and astronomical operations at many stations across the world.

The Field System provides the following capabilities:

- Control and monitoring of various digital backend equipment and recorders.
- Customisation to support site specific equipment and antenna interfaces.
- Provides a suite of programs for antenna pointing calibration and receiver sensitivity measurements.
- Automated scheduling of VLBI operations.
- Logging of events to a log file.

The Field System is well-proven, and its widespread usage, functional capabilities and adaptability are some of the reasons why it is used in AVN.

SNAP

The primary means of interacting with the Field System is via a command-line interface called oprin (operator input) and user input is in the form of SNAP (Standard Notation for Astronomical Procedures) commands [2]. SNAP allows for a high degree of flexibility allowing custom station specific procedures to be created.

Station Software

The Field System provides hooks for implementing station-dependent programs. The station-dependent programs include antcn, stqkr, cheks, and stqkr. The most important of these programs are antcn and stqkr, which allow for station-specific antenna control and monitoring logic to be implemented, and as previously mentioned, custom SNAP commands.

Antenna Control (antcn) The antcn program is customised for the antenna interface at a particular station. It is responsible for all antenna control interactions such pointing, tracking, stow and monitoring the position of the antenna. It is worth noting that antcn does not directly control the antenna but merely provides an interface for the Field system to send requests to software responsible for the control of the antenna.

Station SNAP commands (stqkr) The stqkr program allows station specific versions of field system SNAP commands to be implemented. This is primarily used for controlling a Receiver system's local oscillator frequency and setting noise diodes. Custom SNAP commands have also been added to communicate with Time Interval Counters.

KATCP Interface The Station Software does not directly interface with AVN software components, but rather communicates with telescope systems via KATCP. The Station Software creates a KATCP client connection with the SCS, where the SCS performs the role of a proxy. Both the stqkr and antcn programs effectively translate SNAP commands to KATCP requests that are sent to the SCS. The SCS then forwards the requests to the relevant telescope systems. Sensor readings for feedback are also obtained via the KATCP interface and stored in the Station Software shared-memory variables. The Station Software KATCP client interface is illustrated in Figure 5.

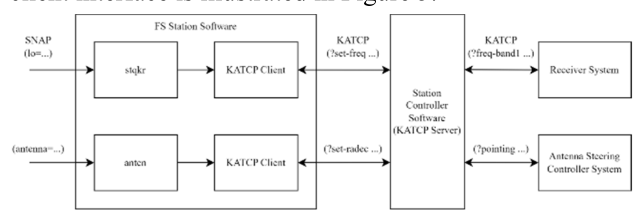

Figure 5: Field System Interfaces.

The Station Software also includes an interface for streaming live Field System event logs to the SCS and processing SNAP command requests via KATCP. This allows an operator to interact with the Field System via the SCS UI instead of a Remote SSH session to the FS Linux server.

VLBI Backend Hardware

The Field System natively supports a variety of the VLBI digital backends for VLBI Operations. The VLBI Backend consists of a Digital Baseband Converter (DBBC) and a Data Recorder.

The DBBC2 manufactured by Hatlab is currently in use in the Kuntunse Station. The DBBC2 is responsible for digital down-conversion of the Analog Intermediate Frequency (IF) signals to baseband frequencies. The digitised and down-converted data is transferred from the DBBC2 to the Mark5B Data Recorder via the VLBI Standard Interface (VSI) and stored on Hard Drive disk packs.

CONCLUSION / ACKNOWLEGMENT

In the conversion of the antenna from a telecommunication station to a radio telescope, we have successfully developed and used software components in the control and monitoring of the telescope subsystems, thereby enabling an additional VLBI resource for the global astronomy community in the pursuit of research science.

The engineering phase of the project in Ghana is intended to be concluded by the end of 2023.

We would like to thank our Ghanaian colleagues for their assistance to make this conversion possible.

REFERENCES

[1] C. J. Copley, *et al* "The African Very Long Baseline Interferometry Network: The Ghana Antenna Conversion", unpublished

[2] E. Himwich, "Introduction to the Field System for Non-Users," in *IVS 2000 General Meeting Proceedings*, Kötzing, Germany, February 2000, paper HIMWICH2, pp. 86-90.