ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA142

THE SKA DISH SPF AND LMC INTERACTION DESIGN: INTERFACES, SIMULATION, TESTING AND INTEGRATION

A. Marassi†, INAF-Astronomical Observatory of Trieste, Trieste, Italy
S. Riggi, F. Schillirò, INAF-Catania Astrophysical Observatory, Catania, Italy
C. van Niekerk, T. Steyn, J. Kotze, EMSS Antennas, Stellenbosch, South Africa
G. Smit, SKA South Africa, Cape Town, South Africa

Abstract

The Square Kilometre Array (SKA) project [1] is responsible for developing the SKA Observatory, the world's largest radio telescope ever built: eventually two arrays of radio antennas - SKA1-Mid and SKA1-Low - will be installed in the South Africa's Karoo region and Western Australia's Murchison Shire respectively, each covering a different range of radio frequencies. In particular, the SKA1-Mid array will comprise of 133 15m diameter dish antennas observing in the 350 MHz-14 GHz range, each locally managed by a Local Monitoring and Control (LMC) system and remotely orchestrated by the SKA Telescope Manager (TM) system. All control system functionality run on the Tango Controls platform. The Dish Single Pixel Feed (SPF) work element will design the combination of feed elements, orthomode transducers (OMTs), and low noise amplifiers (LNAs) that receive the astronomical radio signals. Some SPFs have cryogenically cooled chambers to obtain the sensitivity requirements.

This paper gives a status update of the SKA Dish SPF and LMC interaction design, focusing on SPF, LMC simulators and engineering/operational user interfaces, prototypes being developed and technological choices.

SKA DISH

SKA-MID1 Dish array is composed of 15-m Gregorian offset antennas (Dish element)[2] with a feed-down configuration equipped with wide-band single pixel feeds (SPFs) for the bands 1 (0.35-1.05 GHz), 2 (0.95-1.76 GHz) and 5 (4.6-13.8 GHz) of SKA frequency. The array will consist of 133 dishes plus the 64 MeerKAT dishes, arranged in a dense core with quasi-random distribution, and spiral arms going out to create the long baselines that go up to 200km.

Four sub-elements can be identified in the SKA-Mid1 dish element (see Figure 1): the *Dish Structure (DS)*, the *Single Pixel Feed (SPF)*, the *Receiver (Rx)* and the *Local Monitoring and Control (LMC)*.

The *Dish structure* features the following components: an offset Gregorian reflector system with a feed-down configuration to optimise system noise performance, a fan-type feed indexer at the focal position which allows for changing between the 5 frequency bands by moving the appropriate feed into position, a pedestal providing a RFI shielded cabinet for housing digital electronics and computing equipment hosting other sub-elements' controllers, hardware for antenna movement control and monitoring (Antenna Control Unit or ACU), power distribution to all sub-elements, networking equipment, lightning protection and earthing, cooling ventilation for all the equipment mounted in the RFI shielded compartment itself.

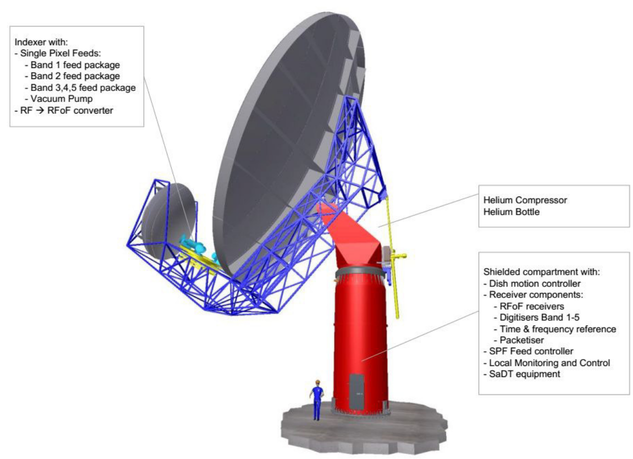

Figure 1: SKA DISH overview.

The *Receiver (Rx)* includes the following components: RF over fibre transport to the antenna pedestal where the digitisers are located, Digitizers performing some RF conditioning (filtering and level control), digitisation, packetizing and transmission to SKA Central Signal Processor (CSP), the Master Clock timer which receives time and frequency reference inputs externally and generates timing and frequency references and a Central controller that acts as single point of control and monitoring to the LMC sub-element.

Dish Local Monitor and Control (LMC) is the subsystem for each dish antenna that deals with the management, monitoring and control of the operation as orchestrated by the Telescope Manager (TM). It consists of a commercial off the shelf controller that serves as a single point of entry for all control and monitoring messages to the outside. Besides configuring the static configurations of the various sub-elements, it also relays the real-time pointing control and applies local pointing corrections. For the monitoring, it aggregates and filters monitoring data as set up from the external (central) controller. The

† marassi@oats.inaf.it

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA142

LMC allows for a drill-down capability for maintainers to access detailed diagnostic information of sub-elements on request. The LMC implements also a circular buffer of detailed monitoring information that can be downloaded remotely for diagnostics purposes after a system failure.

Single Pixel Feed (SPF) sub-element is primarily responsible for converting the electromagnetic (EM) signals focussed by the reflectors to radio frequency (RF) signals that can be digitised. It must amplify these signals while adding minimal additional noise or radio frequency interference (RFI), and allow injecting calibration noise signal.

SPF receivers include feed packages for the bands 1 (0.35-1.05 GHz), 2 (0.95-1.76 GHz) and 5 (4.6-13.8 GHz) of SKA frequency, three cryostat assemblies (respectively for band 1, band 2 and band 3,4,5) housing each a Gifford McMahon (GM) cryogenic cooler to cool the LNAs at a set point of approximately 20K, a second amplification stage and a calibration noise source, both temperature stabilised inside the vacuum, a common shared Helium System, a Vacuum System and a SPF Controller (SPFC), i.e. a single controller located in the pedestal which controls and monitors (see Figure 2) all three feed packages, helium system and vacuum system.

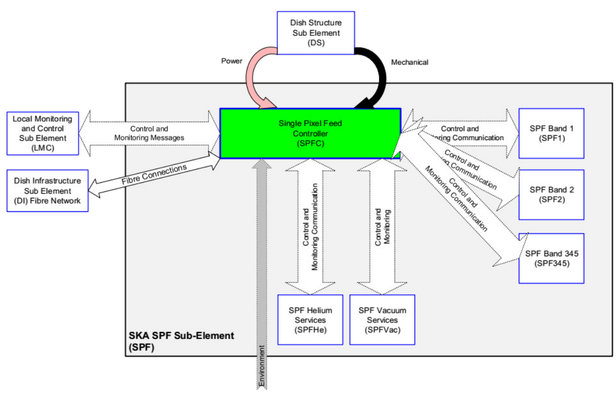

Figure 2: SPFC context diagram.

It monitors the health status of each SPF cryostat, of the receiver electronics and also controls the active components inside each receiver cryostat. All monitored parameters are independently logged per receiver feed package (FP) and sent to the end user when requested. SPFC interfaces with the Dish LMC for external control and monitoring and communicates with it via the Tango protocol over an optic fibre Ethernet interface. ON/OFF control of the low noise amplifiers (LNAs), temperature control set points, and the cool down and vacuum procedures are accomplished via the network interface. Secure shell (SSH) connections to the SPFC are possible in order to change initialization files and update software, while internal log files can be downloaded with an SSH file transfer protocol (SFTP) connection.

Once initialised, the SPFC shall automatically control each connected FP to enter into the user defined default start-up-mode. The SPFC shall control the states and modes of the FPs, SPFHe and SPFVac based on the commands received from the LMC.

Dish Interfaces

Systems interoperate using interfaces. Interfaces are used to support both system to system communication as well as supporting the complete set of enterprise goals.

Monitoring and Control (M&C) operations will be performed on a local network enabling communication between the LMC software system and Dish sub-element controllers. An array fibre network will enable communication between TM and LMC. The communication is physically realized by means of an Ethernet optical fibre link with a network switch.

The functional monitoring and control interfaces with DS, SPFRx and SPF are described in the LMC-sub-element Interface Control Documents (ICDs) ([3], [4], [5] respectively), including communication protocol, exchanged message/data types and format (e.g. commands and responses, monitoring points, alarms and notifications). It was agreed that the application layer shall be in accordance with TANGO (CORBA + ZeroMQ) [6] for LMC-SPF and LMC-SPFRx communication. A custom protocol, with raw TCP sockets as transport layer, is instead adopted for communication between LMC and DS sub-element [3].

LMC, SPFRx, SPF are built upon TANGO device servers as defined in [6]. Control and configuration operations requested by TM in the Dish are performed via a single Tango device server acting as a dish master/interface device.

The interfaces between LMC and other sub-elements, except for DS, are defined in terms of Tango commands and attributes exposed by each device server.

LMC-SPF ICD details interface requirements (as regards configuration and setup, states and modes, sub-element control, fault reporting, diagnostics, alarms and events reporting, remote support) implementation (interaction model, naming conventions, facility context, control model, configuration parameters, commands as regards control and configuration, logging and maintenance, Tango device attributes monitoring as regards states and modes, control and setup, fault and diagnostics, alarms and events handling, archiving, remote support interfacing for software update and engineering GUI) and different use cases views (startup, shutdown, power cut and power restore events, SPF mode control, SPF errors handling, engineering GUI).

Figure 3 shows the identified interfaces (depicted as green boxes) together with the SPF components realizing them (depicted as boxes connected to interface blocks by connectors shown with red dashed lines). Interface users in different systems/subsystems (Dish, Dish LMC, Telescope Manager (TM)) are represented by component boxes or actors connected to the corresponding interface blocks with black dashed lines.

The functional monitoring and control interface with SPF is described in the ICD document [5] which has been

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA142

revised, modified and integrated several times according to continuous requirements refining and to the adopted TANGO control framework specifications.

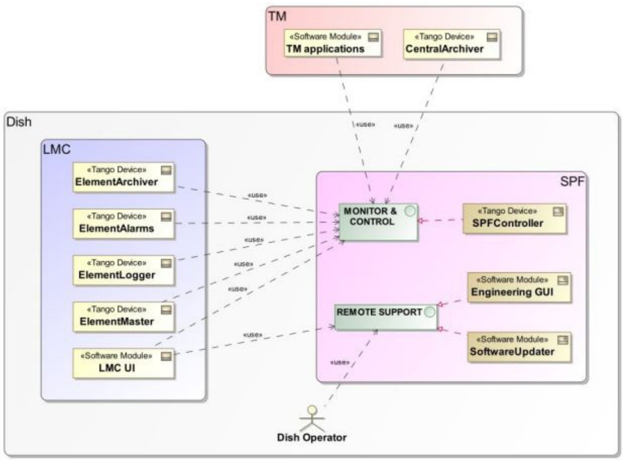

Figure 3: Identified LMC-SPF interfaces, components and users.

SPFC Software Emulators

The use of sub-elements emulators to perform the final integration of devices oriented to the functionalities of the overall DISH.LMC system has been proposed in [7] for LMC integration and standalone qualification (see Figure 4).

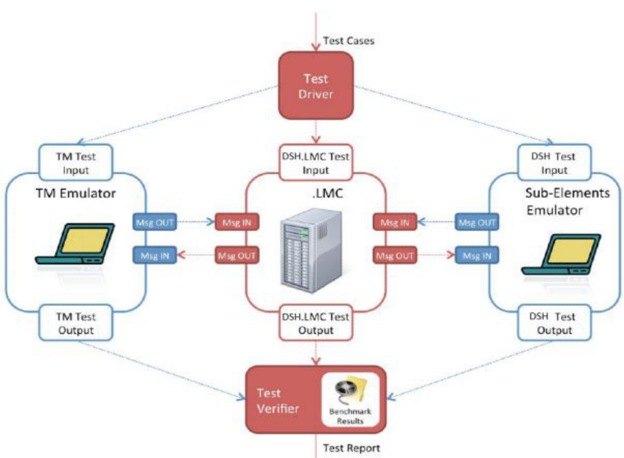

Figure 4: A possible DISH.LMC Test Infrastructure.

Software emulators may be used also to refine interface requirements and to test interaction in simulated real operating conditions.

A basic SPFC sofware emulator, based on LMC-SPF ICD has been developed and deployed on a Tango virtual machine (VM), built with VirtualBox, located in Catania (Italy); it has been interfaced and tested from another Tango virtual machine located in Trieste (Italy) first via simple python scripts and then via a simple prototype GUI, implemented with Taurus tools (Taurusdesigner, Taurusform).

Tango Systems Interfacing

Several tests have been carried out in order to try and interface remote Tango systems located behind firewalls. Tango service running on a host uses a specific port chosen during installation, typically 10000 but, except for the database, Tango uses dynamic allocated port numbers for communicating between clients and servers. This means it is very difficult to know which ports to open in a firewall unless you can open all ports to and from certain hosts. Different solutions have been proposed to this such as the use of the REST API to communicate through the firewall (this is http based and uses only one port), or a proxy which crosses firewall but the latter solution does not seem mature enough. If you use the REST API you cannot use the other APIs (Python, C++ and Java) through the firewall.

Several tests have been carried out in order to try and interface remote Tango systems by setting different firewall ports configurations, both via SSH port forwarding and VPN between Tango VMs located in Trieste (Italy), Catania (Italy) and Stellenbosch (South Africa). SSH port forwarding proved to be awkward and limited (we noticed ZMQ issues), while VPN seemed to solve the problem, apart from some additional latency issues.

Early-Integration Activities

Early tests of the LMC-SPF interface were carried out in June 2017 during a face-to-face meeting between LMC and SPFC teams, using software emulators based on LMC/SPFC ICD. The testing infrastructure (see Figure 5) was based upon a switched network which connected three PCs running VMs simulating respectively SPFC, DISH.LMC and an LMC engineering GUI to be used for maintenance, debugging and integration purposes. It is to tunnel also other DISH sub-elements engineering interfaces.

This first prototype basic GUI to monitor and control SPF parameters, organized according LMC-SPF ICD was built with Taurusform tool, after having been simulated via throw-away mock-ups implemented as an interactive PDF and used to get feedback from end users and stakeholders .

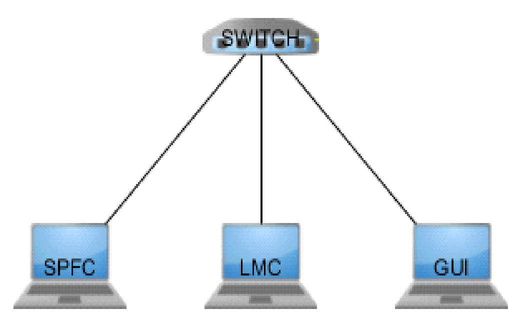

Figure 5:LMC-SPFC pre-integration testing infrastructure.

The performed early-integration activities have greatly contributed to dish LMC-SPF interface update.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA142

They are to be considered as an important effort in sub-elements interface definition and verification towards Dish integration as defined in [7] and [8].

The same approach will be soon adopted with the other two Dish sub-elements: DS and SPFRx.

DISH INTEGRATION, VERIFICATION AND QUALIFICATION

During the systems integration process, you take the independently developed subsystems and put them together to make up a complete system. Integration can be done using a 'big bang' approach, where all the sub-systems are integrated at the same time. However, for both technical and managerial purposes, an incremental integration process where sub-systems are integrated one at a time is the best approach, for the impossibility to schedule the development of all the sub-systems so that they are all finished at the same time and also because incremental integration reduces the cost of errors location in case many sub-systems are to be integrated [9].

Once the components have been integrated, an extensive programme of system testing takes place, both aimed at testing the interfaces between components and the behaviour of the system as a whole.

Introducing some SKA.DISH terminology [8]:

Verification may be defined as the confirmation, through the provision of objective evidence, that specified requirements are fulfilled.

Qualification is the process of verifying that a specific design definition (part number) fulfils its requirements, typically by testing a pre-production model. Qualification is required before commencing with production to ensure that the risk of producing non-compliant items is sufficiently low. A qualification program is of particular importance if a large amount of units will be produced with a long expected service capability. Qualification testing is typically a comprehensive test of all specifications, including environmental testing.

Acceptance verification is performed on all production items to ensure that they meet all requirements, and that the quality of the production process is under control. Because the design is already qualified, the acceptance testing is typically a subset of the qualification tests.

Integration is the process of successfully combining hardware and software components, sub-systems, and systems into a complete and functioning whole.

Integration is an iterative process:

- taking hardware and software components
- forming them into complete sub-system elements
- combining the sub-system elements into larger combined sub-systems
- combining all sub-systems into the final system

The main activities on the Dish element level and sub-element level teams relating to integration and verification are: *Sub-element Detail design* based on the sub-element requirements, whose maturity has to be accessed by a detailed design review. A sub-element qualification plan is prepared as well, that spells out the qualification events, activities and procedures for the sub-element. *Sub-element Qualification* in which the sub-element work group starts the process of constructing the qualification model (QM). The QM is then integrated at sub-element level and subjected to a rigorous set of qualification testing to verify that the design meets all the sub-element requirements. *Dish Element Qualification:* once sub-elements are qualified, they are integrated to form a complete Dish system (Dish structure, feeds, receiver, LMC, etc.). The integrated dish is then tested to verify that it meets all the Dish system requirements.

SUMMARY

The SKA project has recently entered Critical Design Review stage of pre-construction in which final software design and qualification are expected in mid 2018 before proceeding to construction phase.

Considerable efforts have been dedicated to LMC software and dish interfaces update also to ensure a high-degree of compliancy with the still-evolving SKA standards and with the SKA adopted Tango controls framework. Connection tests have been carried out between remote Tango systems located behind firewalls by setting different firewall port configurations, and by using SSH port forwarding or VPN.

Prototype GUIs to monitor and control SPF parameters, organized according LMC-SPF ICD were built using different Taurus tools after having been simulated via throw-away mock-ups implemented as an interactive PDF and used to get feedback from end users and stakeholders.

Software implementation is in advanced status for some components of Dish sub-elements while others are awaiting for progresses in design and interface consolidation. In order to overcome different components maturity, for interface definition and testing purposes device emulators have been developed for SPF sub-element and early tests of the LMC-SPF interface were carried out in June 2017. The same approach will soon be adopted with the other two Dish sub-elements: DS and SPFRx and possibly with TM element.

REFERENCES

[1] Square Kilometre Array, http://www.skatelescope.org

[2] SKA DISH System Overview , SKA-TEL-DSH-0000017

[3] Data Exchange Interface Between The Dish LMC and the DS Controller, SKA-TEL-DSH-0000053

[4] Data Exchange Interface Between The Dish LMC and the SPFRX Pedestal Unit, SKA-TEL-DSH0000054, Rev 1

[5] Data Exchange Interface Between The Dish LMC and the SPF Controller, SKA-TEL-DSH-0000055, Rev 1

[6] The TANGO Control System Manual, Version 9.2

[7] DISH.LMC Integration and Qualification Plan, SKA-TEL.DSH-320-000000-001 Rev.B

[8] SKA DISH Element Integration and Verification Plan, SKA-TEL-DSH-0000024, Rev. 2

[9] I. Sommerville, Software Engineering Eighth Edition Addison-Wesley 2007