**Mission Thread Normal Observation**

**Introduction**

This mission thread focuses on the SKA Org’s efforts to develop and convey...

**Supporting Diagrams**

(Insert at least a context diagram. Sometimes it is helpful to include other diagrams like a “future vision” or “legacy” system, but don’t want to clutter this area up with a lot of pictures because it becomes a distraction.)

|  |  |
| --- | --- |
| Name | SKA Normal Observation Operation |
| Vignette / Summary | An observation proposal is submitted and accepted. The proposal is used to plan and execute an observation, data is acquired, and science data (along with any other relevant metadata) is produced and stored.  For simplicity, this thread considers an observation that consists of 2 scans.  This thread does not consider Targets of Opportunity – that will be defined and analyzed as an extension to this thread. |
| Nodes/Actors | 1. Telescope Manager (TM) 2. Central Signal Processor (CSP) 3. Science Data Processor (SDP) 4. Signal and Data Transport (SaDT) 5. Array (Mid and/or Low) |
| **Assumptions** | 1. This is targeted at SKA-low. 2. xxx 3. xxx 4. xxx   Questions:   1. Do we need input X for proposal (and what is not needed)? 2. xxx?   Notes: TPM = Tile Processing Module.   1. xxx |
| **References** | 1. SKA Low Functional Architecture Document 2. SKA Mid Functional Architecture Document   Telescope Observing Control View working paper |

**Mission Thread Normal Ops - Steps**

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Description | Inputs/Outputs | Issues |
| 1 | Proposal accepted | Out: Technical data from proposal | Are we omitting proposal eval. Decoupled from observation. Proposal outputs will be mandated by what is needed in obs. Have to do proposal assessment based on prob. assessment of what resources will be available (and minimums needed). Scientist will not say “I need this TPM” (ask what they want and translate). In ALMA this is the translation between science goal and technical goal. Scientists should specify minimal quality. In MWA people generally do not want control over ‘knobs’. Easier to define observation based on least you can accept. Scientists guess certain # of hours given, and work backwards from likely resource available (based on likelihood of acceptance) |
| 2 | Design Observations | Out: Scheduling Blocks | SB are Units of planning rather than execution. Unit of execution is a scan. A diagram/model to capture relationships between domain concepts (SB/scan) useful. SB script. SB is indivisible(?). Distinction between config, execution, planning. SB is meaningful from a science perspective. Calibration is not necessarily part of SB. Few hundred science proposals per year. SB time ~ 1hr. (TBD) About 10k SBs per year. (more with commensal obs). Need to know ‘resources’ required. There is a set of existing designs (e.g. LFA array) with existing configurability. Arbitrary constraints on what seems like infinite config. (since many configs nonsensical) |
| 3 | Plan Observations | Out: Observation plan | Needs a projection of available resources (downtime etc, evolving resources (more telescope)). Constraints on sky, weather. Need to resolve science view vs operational view. See ICD – field node/station. Specify tile # and lat/long. Need to map ICD onto science requirements. Some rules exist for doing the observations . Part of this plan is choosing ideal set of resources (which can change at config time) (or matching the set of requirements to realistically available) |
| 4 | Configure subarray |  | In principle … a tile can be used in multiple subarrays(?) (and per freq. blocks) SDP must be able to handle more complex data flows. Finest-grain is continuous stream of data (and so unaware of more complex abstractions) but costly (esp for correlator and SDP). May be very complicated to re-use tiles in different subarrays. Dominated by processing bandwidth (300Mhz). What is the schedulable resource. – a station is (set of tiles/..) At what point do we care about the freq. independence of the stations. What is corresponding resource to ‘dish’ for Low. At same time as you do core data for SDP process you have to do long baseline (EOR) work. Why separate subarray config. Observation config is more detailed config. Subarray config is about allocation of resources. This is TM doing ‘bookkeeping’. Configurability is overloaded. The following steps are parallel. Need to know both current state of subarray and then planned future. Level of abstraction of configuration (specific elements vs system).  Subarrays are completely independent entities. Commensality makes this unrealistic. |
| 4.a | Allocate resources |  | Define allocate. Set apart. Reserved / assigned / exclusive use. (in theory not true for low). In practice? At one point in time we can do multiple observations. Build a datastream (make a pipeline/connection vs assign resources) Without connections the allocation is not meaningful. Allocation not covered in requirements. What are subarrays doing. System state relevant in seconds. Just do a check pre-execution of SB. Subarray will not change, but losing a resource might affect the ultimate data quality (sensitivity etc) |
| 4.a.i | Allocate Tiles | Out: Station definitions to LFAA | Resource allocation – must allocate entire tile (but can mask out to get one antenna). |
| 4.a.ii | Allocate CSP input bandwidth | Out: IP addresses to LFAA |  |
| 4.a.iii | Allocate cross-correlation matrix |  |  |
| 4.a.iv | Allocate SDP ingest buffers | Out: IP addresses to CSP |  |
| 5 | Select Scheduling Block | Out: Scheduling block instance, translated for TelMgt | May not be same representation of SB in TM and databases. “Invoke” block on TM. Resource avail. important. Can’t plan for a specific resource far in advance. 2 process running concurrently: TelMgt and ops manager. (per subarray). Abstraction on physical antenna or datastream (for every freq channel have your own ‘dish’). Known hours ahead. SB might be a script and database that get transported. Not having everyone access a db is easier. Tradeoff. Online and offline different. The online stuff also dealt with OpsMgt. State maintenance important (where stored, who accesses, consistency). Physically extended system means latency of state is key. Decision on resource allocation is one system’s (TM) responsibility. Scheduling happens at different times. Capabilities are not changeable except by TM. What system looks like at moment – (what is at the moment, secs). This time interval affects speed of configuration. 2 timescales. e.g. morning do day’s plan, also few secs before just to check. When should detailed/general/ planning be done (and for what subarrays (e..g low/high sensitivity)) Is there a cost implication to changing subarray config (as well as a time config). Do you need advanced planning (you do for maintenance, and for overcommitting resources). Sky is a resource. On-demand/JIT planning. How stable is the system incl. physical constraints)  RFI environment relatively stable (sat, TV,radio etc). Can be planned but is dynamic  Ionosphere fairly unpredictable  These sources of error are detected by different components in the system (SDP vs TM?)  Latency is quite low (fuzzy). Most failures aren’t going to ruin an observation.  ALMA does JIT observation planning (hi-freq) based on priority ObsMgt makes decision on what scheduling block to execute. Invokes TelMgr seconds before (a plan of SBs). Has flexibility to remove/add SBs. A human makes the decision for what is in queue. Each subarray has own queue. (OD subcomponent of ObsMgt)  Items in different consortia end up closely physically located |
| 6 | Configure Scheduling Block |  | Send commands (in parallel) to subsystems. Separate scan-dependent from scan-independent.  Scan is a run – take data.  Commands given to TelMgr don’t change between scans.  Params at beginning of a scan, and some during a scan  A script specifies the full setup.  A SB config is complete. It should know the state of system calibration  Obs design needs to specify calibration needed.  Is there an observatory-specific scheduling block just for calibration.  Standard calibration ‘button’ for an observation.  Does ICD define when params change (and which ICD does that)  What is Completeness (and when) should be defined architecturally. Could be flagged in pipeline and completion managed in SDP. Full vs incremental config set.  Should the system wait for all the following steps be completed?  Do we get a response from a dish to say it is on new position?  How long do we wait til it should be there? When do errors get reported  ObsMgr needs to know this  The need to flag data is a work-around for ineffective control system  Assume everything goes well until it doesn’t  Protocol separates command from control  Tradeoffs between how/who to decide success/failure (TM makes decision, and treatment of all elements is consistent)  Never clear to the element if a failure is critical or not (only TM knows this)  Knowing mean time to complete per element helps with reporting  Do we report status or judgment? Distinguish between reporting status in the architecture.  LMC will report an alarm (e.g. lost a baseline) and telescope operator will report that and carry on  Does TM know about science? The ObsMgt component does. TelMgt does not. TM is same for Lo and Mid (same people) |
| 6.a | Configure LFAA | In: beam phase centers (RA/dec), beam coarse channelization, beam weights, antenna weights, beam-to-channel mapping, firmware to load (FPGA). | 512 datastreams (inputs to correlator). Coarse channels fixed.  correcting array differences should happen at receiving end of beams (CSP). Sum in LFAA is formed in multiple places.  3 sets of beam weights (station, array, pulsar processing session)  Beam weights large # of values but not large amount of data. Need to send core + alg. to each station  Pattern: compute centrally + send results, or distributed alg. + reduce centrally. Be consistent. Depends on relative size of input / output.  Imposes some domain knowledge on TM  Haven’t defined what software for algs (beam weights) is (language, classes etc) key to the architecture. Only then can we figure out where to put that software.  Proj management point of view TM/CSP not clear who owns this software (calcsolve, Kat point). TM is responsible for core pointing/platform software. Need interferometry experts who TM doesn’t have.  Reflash to new firmware.  Where does latest version of firmware live and . TM owns the system and config mgmt. of all software. Separate from TM itself.  2 types of s/w: small # of instances (TM-low etc) and s/w with instances per instrument (8192 or so (per instrument))  Who makes the change in s/w. Evolution mechanism.  Useful to know what version triggered a bug  TPM level of firmware is large number. 8192 separate installs. Latency over distribution. Should the beam to channel and firmware mappings be in subarray level  What ends up in science data product and what is in archive  These commands are at sub-array level  Sub-array min lifetime is scheduling block. Is there a penalty to specifying sub-array config each time?  Useful to human team to know. No problem to CSP. Often a no-op. Param space is more complex for scheduler.  Subarray to TM means whole system (CSP and all schedulable resources across Telescope), while something smaller to beam former.  Scheduling can be asynchronous so subarrays get dependent (grabbing resources)  Up to 8 beams per station  1 station per subarray. All beams of a station part of same subarray.  Clarify limit imposed by correlator (512 streams w/ 64k channels) vs raw capability of system |
| 6.b | Configure CSP | In: imagining phase centre, imaging chanellization, Imaging dumput tie, PSS channelization, PSS beams (inputs, phase centres, PST channelization, PST beams (inputs and phase centers). |  |
| 6.c | Configure SDP | In: Ingest buffer size (?), Processing templates (capability) names |  |
| 7 | Ready for observation | Out: signal capture control, signal capture configuration, signal processing control, signal processing config. |  |
| 8 | Configure Scan |  | Change only what changed between scans. |
| 8.a. | TM | Out: Scan ID, Scan start time |  |
| 9 | Scanning |  | During a scan |
| 9.a | LFAA scanning | In: Geometric/Ionispheric Antenna delays/Jones matrices. Out: Station observation RF data, QA metadata (RFI flags etc) | What does it get for input? |
| 9.a.i | Generate instrumental delays | In: local sky model |  |
| 9.b | CSP.CBF scanning | In: Station Jones matrices (?), Beamforming weights (delays), Transient trigger Output: Cross correlations, Transient buffers | Not clear who produces Jones matrices etc (are these functional models or outputs from the models). Partly output from SDP (delay models/tracking). Is this necessary. Only for pulsar. Cadence of 10s. CSP generates error correction. Should SDP feedback directly or go through TM. What transport size. |
| 9.c | CSP.PSS scanning | In: Beam weights, Beam Jones Matrices Out: Pulsar search candidates, Pulsar timing QA |  |
| 9.d | CSP.PST scanning | In: Beam weights, Beam Jones Matrices Out: Pulsar timing observations, Pulsar timing QA metadata | Come from SDP through TM for each individual pulsar. Why go through TM? (1Gbit/s). RFI and Zoom are configured in between scans (not realtime).  Do these Jones matrices need to be stored? Need to be seen by anyone else (debug)?  System requirement that no 2 systems talk to each directly  TM is responsible for synchronization (expected of TM)  Involving TM, have 3 consortia not 2  Tango – P2P communications broker. Tango is mandated by TM (part of). Tango reads p2p info from db and acts as broker.  Impacts ICDs.  Need to document (given choice of pattern) what QoS guarantees are important and why  Can’t solve problem by re-transmit – have to just carry on  Correlation and beam-forming happen together  Need to have single way of doing things across products/consortia.  Is Tango (pipes) good enough for this transmission (vs pipes/shared mem etc)  CSP responsible for reporting errors/flags  Interface docs need to express producer/consumer relation  High level Temporal sequence and connection docs missing. ICD describes the ports. |
| 9.e | SDP scanning | In: upstream data and metadata, Observation QA, Output: Transient information, Ionospheric models, Local sky model | SDP realtime pipeline incl to CSP |
| 10 | Ready (Observation not complete) |  |  |
| 11 | Configuring (for second scan) |  |  |
| 12 | Ready (for second scan) |  |  |
| 13 | Scanning (second/final scan) |  |  |
| 14 | Ready | In: all scans for this observation complete |  |
| 15 | Idle | (SB complete) | Acquisition complete. Why separate SB from observation. |
| 16 | Process science data | In: Science data, Telescope status data Out: Processed science data, User notifications, QA Information | Long life-cycle (process data from 10 days ago). ‘Reset SDP’ means what? This step only covers the ingest buffers.  Result of processing SDP blocks has impact on scheduling.  What is QoS on constraints on data/SB scheduling  What subystems are affected by SDP lag? |
| 17 | Deliver science data | Out: Science data, QA data, Other meta data |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| Extensions |  |  |  |
| Ext. 3.1 | External partner executes the recommended action. |  |  |

# 

| Quality  Attribute | Overarching (end-to-end) Considerations, Issues, Challenges |
| --- | --- |
| Interoperability |  |
| Performance |  |
| Availability |  |
| Security |  |
| Capacity |  |
|  |  |
|  |  |

# Attendees (x-attended)

## Imports

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |