Software - SSS

Bryan Butler
Major Subsystems

- *High Level Architecture (HLA)*
- Portal (user authentication, and gateway)
- Proposal preparation, submission, handling (PST)
- *Observation preparation (OPT)*
- *Array Scheduling (OST)*
- Archive access (AAT)
High Level Architecture

- The major subsystems are known (and have been for several years)
- The main task is now in defining the interfaces - we define these as “models” (they are really objects) used within, and passed between, subsystems
- Significant progress has been made in the last 18 months in developing these models, which are common throughout the system as much as they can be, so reused by multiple subsystems
- A caveat is that with distributed development (EVLA, ALMA, EOD), use of “common” models means something different than one might think or hope
Example - Project Model

Location of Resource in Project Model

Telescope

TelescopeType

VLA_A = "G": The Very Large Array
VLA_B = "R": The Very Long Baseline Array
VLA_C = "T": The Expanded Very Large Array
VLA_E = "L": The Atacama Large Millimeter Array
VLA_S = "E": Other
VLA_X = "U": Unknown

Receiver

Set of Receiver

Resource

Source

ScanLoopElement

ScanLoop

SchedulingBlock

ProgramBlock

- project

List of ProgramBlocks

Project

Telescope

location

TelescopeConfiguration

List of TelescopeConfigurations

2007-Sep-06
The Resource Package represents the selection and configuration of the hardware used to make an observation. The hardware is represented above by the Receiver and TelescopeBackend classes. The ResourceSpecification class represents a science (i.e., hardware-neutral) view of the Resource. This specification is optional. When present it can be used by intelligent classes (like ResourceSelector, BackendSelector, etc.) shown here to select hardware and (WidderConfiguration) to configure hardware.
Observation Preparation (OPT)

- Converts Project into Program Blocks (PB is a collection of Scheduling Blocks)
- Needs detailed telescope knowledge
- Needs to support 3 “levels” of user:
  - Novice (automatic generation of PBs for “standard modes”)
  - Intermediate (graphical setting of observing parameters)
  - Expert (allow for script level editing)
OPT - Components

- Authenticate
  - PdB
  - Read in PDM
  - Best Guess for PBs & SBs
  - Modify PB

- Exit
  - Write to PBdB
  - Validate

- PBdB
OPT - Components

Modify PB

- Create/Modify an SB
- Add an SB to the PB
- Remove an SB from the PB
- Modify SB Contingencies
- Modify PB Constraints
OPT - Components

Modify SB

In current OPT, but minimal

Not in current OPT

In current OPT
OPT - Demo

NRAO > User Portal > Observation Preparation Tool
> Source Catalog Tool
> Resource Catalog Tool

Return to Uranus

Scheduling Block Details

<table>
<thead>
<tr>
<th>Name</th>
<th>Uranus - K-band</th>
<th>Status</th>
<th>Not Yet Scheduled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>0</td>
<td>Total Time</td>
<td>0.0 HOUR</td>
</tr>
<tr>
<td>Num. Times Completed</td>
<td>0</td>
<td>Time Per Execution</td>
<td>0.0 HOUR</td>
</tr>
<tr>
<td>Num. Times Aborted</td>
<td>0</td>
<td>Monitoring Interval</td>
<td>0.0 HOUR</td>
</tr>
</tbody>
</table>

Consecutive Iterations? [ ]

Scans

- Standard Observing on 'J246-1206' for 1.0 MINUTE
- Standard Observing on 'Uranus' for 5.0 MINUTE
Observation Scheduling (OST)

- Takes pool of available Scheduling Blocks and chooses what should be observed next on the telescope
- Must take into account scientific priority, current observing conditions, and other priority modifiers (project completion pressure, for example)
- Must be able to run autonomously, eventually
OST - Components

SBDB → Read in SBs → Calculate Priorities → Present Ordered List to Operator → Modify SB and write to DB

check LST → check contingencies → get scientific priority → get environmental modifier → get operational modifier

SB in → Input from M&C → SB with modified priority out

“Program Manager”
OST - Tests on VLA

- Prototype system tested on the VLA - in current regular use (ancestor used successfully for VLBA for several years)
- Observers send in modified OBSERVE files (extra “header” information included in comments at the top)
- OBSERVE files wrangled by NRAO staff
- Prototype OST system presents operator with prioritized list of what to observe next
- All command line tools
- Providing us with invaluable information on the practical aspects of dynamic scheduling of a many-element radio interferometer
OST VLA Tests - Lessons Learned

- It works! Fundamentally, the VLA can be dynamically scheduled, with Scheduling Blocks drawn from a pool and subsequently observed.
- The system is inordinately fond of short SBs - it works well for them, but medium to long length SBs (≥~ 2 hours) have not been well tested.
- Popular LSTs are efficiently used - those LSTs that are not so oversubscribed are not so efficiently used.
- Currently effort-intensive (but getting better)
New tool which fits in to the HLA was developed during late 2006 to mid 2007

- GUI - much easier to use than old CLI

- Provides framework for testing many different heuristic schemes

- Intent was to deploy for testing at VLA summer 2007, but primary developer has left NRAO, slowing this down (replacement employee starts Sept. 10)
OST - new GUI tool