Status of ASKAP Control System

Juan Carlos Guzman
ASKAP Software Engineer – Control Software Lead

11th October 2010 – EPICS Collaboration Meeting, BNL, NY
ASKAP Project
Australian SKA Pathfinder = 1% SKA

- Wide field of view radio telescope
  - Sited at Boolardy, Western Australia
  - Observes between 0.7 and 1.8 GHz
  - 36 antennas, 12m diameter, 3-axis
  - Phased Array Feed (PAF) technology (36 beams)
  - 30 sq degree field of view
  - 6 antenna prototype mid 2011 - Boorlardy Engineering Test Array (BETA)
  - Full system mid 2013

- Scientific instrument, optimised for survey
  - Survey HI emission from 1.7 million galaxies up z ~ 0.3
  - Deep continuum survey of entire sky
  - Polarimetry over entire sky

- Technical pathfinder
  - Demonstration of WA as SKA site
  - Phased Array Feeds
  - Computing
Murchison Radio Observatory (MRO): Australia’s SKA Candidate site

Traditional lands of the Wajarri Yamatji
Geraldton

Gazetted towns: 0
Population: “up to 160”
# ASKAP Project Timeline

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 2010</td>
<td>First antenna installed at the site</td>
</tr>
<tr>
<td>May 2010</td>
<td>Subsystem’s CDR completed</td>
</tr>
<tr>
<td>Mar 2011</td>
<td>Installation of Full-PAF + new hw/sw backend at Parkes</td>
</tr>
<tr>
<td>Mar 2011</td>
<td>SAT of antenna 2 – 6 (BETA) completed</td>
</tr>
<tr>
<td>May 2011</td>
<td>Fibre-link between Geraldton and MRO complete</td>
</tr>
<tr>
<td>Jun 2011</td>
<td>Installation of Full-PAF+hw/sw back-end on BETA begins</td>
</tr>
<tr>
<td>Aug 2011</td>
<td>Early commissioning of BETA begins</td>
</tr>
<tr>
<td>Oct 2011</td>
<td>MRO infrastructure complete</td>
</tr>
<tr>
<td>Jan 2012</td>
<td>Installation/SAT of antennas 7 – 36 complete</td>
</tr>
<tr>
<td>Nov 2012</td>
<td>36 ASKAP antennas with PAF + hw/sw installed</td>
</tr>
<tr>
<td>Dec 2012</td>
<td>ASKAP “Early Science” (commissioning/shared risk)</td>
</tr>
<tr>
<td>Jul 2013</td>
<td>Initial Science Survey Projects observing</td>
</tr>
</tbody>
</table>
ASKAP Site
ASKAP Site
ASKAP Data Flow

**ASKAP FPGA-based signal processing chain**

- **Thirty six antennas**
  - Filterbanks
    - 18Tflop/s
  - Beamformers
    - 27Tflop/s
    - PAF filterbank samples
    - 1.9Tflop/s

- **MRO**
  - 0.6Tflop/s
  - Beamformed filterbank samples

- **Correlator**
  - 2.5Gbps

- **Central processor**
  - 100Tflop/s

- **Operations data archive**
  - 100Gb/s

- **Pawsey Centre**
  - ASKAP Science Data Archive Facility
  - ASKAP products via VO protocols

- **Virtual Observatory**

---

Status of ASKAP Control System - EPICS Collaboration Meeting, BNL, NY, USA, Oct 2010

T. Cornwell, Feb 22 2010
ASKAP Control Software Architecture

Enterprise Service Bus = ZeroC ICE
EPICS IOC Implementation

• Using EPICS base 3.14.11
  • Support (development) for Linux (x86_64) and MacOSX (Snow Leopard)
  • No need for real-time OS extensions for now

• Estimated number of records ~ 180,000 (~ 5,000/antenna)

• Estimated number of deployed IOCs ~ 40
  • SoftIOC on Linux (non-real time) OS
  • We call them Control Computers
  • 1U rack-type computers running standard Linux (debian) distro x86_64 type (diskless or solid-state disk)
EPICS IOC Implementation

• Design and write the EPICS database (list of EPICS records). Use of ICD as reference
  • Keep IOC database as simple as possible
  • Extensive use of MSI
• Write SNL programs for some applications (version 2.0.12)
• Extensive use of EPICS ASYN framework (version 4.14)
  • Code needs to be added if field-bus protocol requires additional ASYN interfaces, i.e. driver support for the ASKAP Digitiser (UDP-based protocol)
  • Driver support can be re-used. We are planning to use same ASKAP digitiser driver support for LO, Beamformer and Correlator
• Implement an emulator (Python)
  • Essential for continuous integration and testing of high-level software components
  • Most of our hardware devices are Ethernet-enabled
• Implementation of EPICS IOC applications are done mainly by Computing Team (Control Group), but in some cases are shared development, i.e. Analog Subsystem

Status of ASKAP Control System - EPICS Collaboration Meeting, BNL, NY, USA, Oct 2010
## EPICS Implementation Status

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Implementation Status</th>
<th>Fieldbus/Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna Drives</td>
<td>50 %</td>
<td>Ethernet/TCP (ASKAP) and Modbus/TCP (Parkes 12m)</td>
</tr>
<tr>
<td>Pedestal and Prime Focus Analog Electronics</td>
<td>50 %</td>
<td>Ethernet/UDP(SPI)</td>
</tr>
<tr>
<td>LO</td>
<td>50 %</td>
<td>Ethernet/UDP</td>
</tr>
<tr>
<td>Timing</td>
<td>Completed</td>
<td>Ethernet/UDP/PCI</td>
</tr>
<tr>
<td>Digitiser</td>
<td>Completed</td>
<td>Ethernet/UDP</td>
</tr>
<tr>
<td>Beamformer</td>
<td>0% ICD in progress</td>
<td>Ethernet/UDP</td>
</tr>
<tr>
<td>Correlator</td>
<td>0 %</td>
<td>Ethernet/UDP</td>
</tr>
<tr>
<td>Power and Cooling</td>
<td>0 %</td>
<td>Ethernet/TCP(Modbus)/SNMP</td>
</tr>
<tr>
<td>MRO Weather Station</td>
<td>0 %</td>
<td>Ethernet/TCP</td>
</tr>
<tr>
<td>MRO Safety (Fire alarms, interlocks)</td>
<td>0 %</td>
<td>Ethernet/ModbusTCP</td>
</tr>
<tr>
<td>MRO Networking Hardware</td>
<td>0 %</td>
<td>Ethernet/SNMP</td>
</tr>
</tbody>
</table>
EPICS Clients Implementation

• Using our own Archiver
  • ASKAP will use software called MoniCA which is used for the Monitoring Archiver role at all other CASS/ATNF Observatories
  • Implemented in Java
  • Supports MySQL and ASCII database backends
  • EPICS CA support via JCA library
  • Google “open-monica” or http://code.google.com/p/open-monica/

• GUIs
  • Started using EDM but there are some issues on x86_64 and MacOSX
  • Converting screens to Qt 4.6 + epicsqt library (Australian Synchrotron)

• Test scripts
  • Implemented in Python
  • EPICS CA support via cothread 1.15 (Diamond)

• VirtualTOS (vtos)
  • Stand-alone package with SoftIOC emulating antenna IOCs + High level applications for creating/executing observations
Build Infrastructure

Features

- Revision control system: Subversion
- Support for C/C++, Java and Python for our code
- Platform independent
  - Compiles in Linux (Debian) and MacOSX
  - Deploy only Linux
- Automated building (and deployment) of dependencies
  - A single build command (rbuild)
    - Updates from subversion repository
    - Recursively builds a package's dependencies
    - Written in Python
  - The dependencies.default file lists a packages dependencies
    - To avoid dependency loops have a simple dependency structure
  - Wraps several “make” tools: autotools, epics makefiles, scons, ant and setuptools
    - Integrate lots of 3rd party software as part of our build
    - Standard patching procedure and option settings
    - Support parallel builds – to speed up a little bit
  - Wraps documentation generation tools: doxygen and Sphinx
  - Can build and run unit and functional test
Switched from Trac to Redmine late 2009

- Redmine: http://www.redmine.org
- Open source project management web application written in Ruby on Rails
- Allows sub-projects, so it is used by ASKAP
- Use the concept of issues (task, bug, feature, etc.) and milestones
- Wiki
- Linked to source repository (SVN, CVS, Git, Mercurial, etc.)
- Email notifications
- Time tracking (although we don't use this)
Continuous integration

• Adopted continuous integration process in July 2009
  • Using Hudson open-source tool: https://hudson.dev.java.net/
  • ASKAPsoft codebase is tested continuously and automatically
  • Each commit to the Subversion repository results in a build/test job being spawned
  • Immediate productivity improvements obvious
    • Reduced time to detect defects (particularly regressions) from days/weeks to hours
    • Easily able to identify exactly which change-set caused the problem

• Trend test additions, deletions, passes and fails over time
Continuous integration

• Hudson dashboard shows status of all build executors and jobs
Square Kilometer Array (SKA)

• Download the SKA animation movie from http://www.skatelescope.org/video/SKA_Animation_2010.mov
  • Credit: Swinburne University
<table>
<thead>
<tr>
<th>Year Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007 – 12</td>
<td>Telescope design and cost</td>
</tr>
<tr>
<td>2012</td>
<td>Site selection (Australia or Southern Africa)</td>
</tr>
<tr>
<td>2012 – 13</td>
<td>Establish SKA organisation + initial (10% SKA Phase 1) construction funding approval</td>
</tr>
<tr>
<td>2013 - 18</td>
<td>Detailed design &amp; construction of Phase 1 at low and mid frequencies</td>
</tr>
<tr>
<td>End 2016</td>
<td>Construction funding approved for 100% SKA at mid and low frequencies</td>
</tr>
<tr>
<td>2017 - 22</td>
<td>Finish construction at low and mid frequencies</td>
</tr>
<tr>
<td>2019 -&gt;</td>
<td>Science operations with Phase 1</td>
</tr>
<tr>
<td>2023</td>
<td>Full operations at low and mid frequencies</td>
</tr>
<tr>
<td>2013 - 22</td>
<td>High frequency technology development</td>
</tr>
<tr>
<td>2023 -&gt;</td>
<td>Start construction of SKA at high</td>
</tr>
</tbody>
</table>
Thank you