Parallel and distributed processing for ASKAP

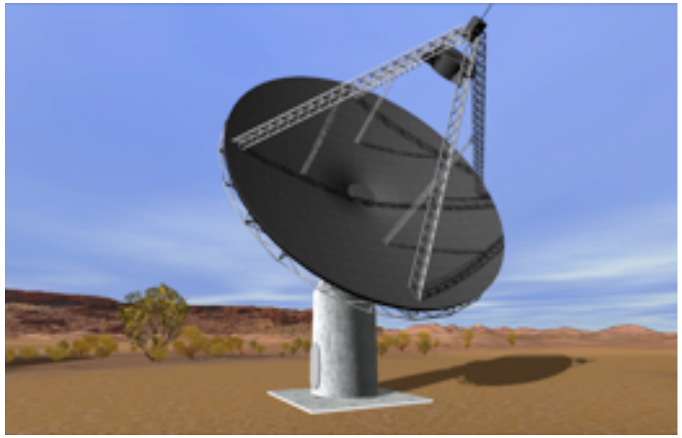
Tim Cornwell
ASKAP Computing Lead
Ben Humphreys
ASKAP Computing Engineer

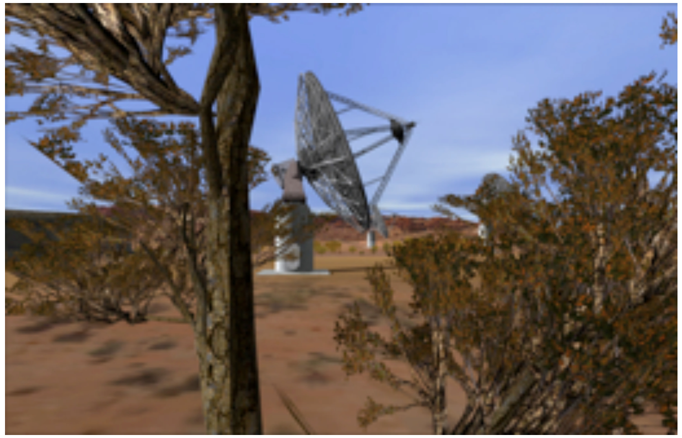
Outline

- ASKAP
- Central Processor
- Single Digital Backend

Australian SKA Pathfinder = 1% SKA

- Wide field of view telescope (30 sq degrees)
  - Sited at Boolardy, Western Australia
  - Observes between 0.7 and 1.8 GHz
  - 36 antennas, 12m diameter
  - 30 beam phased array feed on each antenna
  - Started construction July 2006
  - 6 antenna prototype (BETA) late 2010
  - Full system late 2012
- Scientific capabilities
  - Survey HI emission from 1.7 million galaxies up z ~ 0.3
  - Deep continuum survey of entire sky ~ 10uJy
  - Polarimetry over entire sky
- Technical pathfinder
  - Demonstration of WA as SKA site
  - Phased Array Feeds
  - Computing

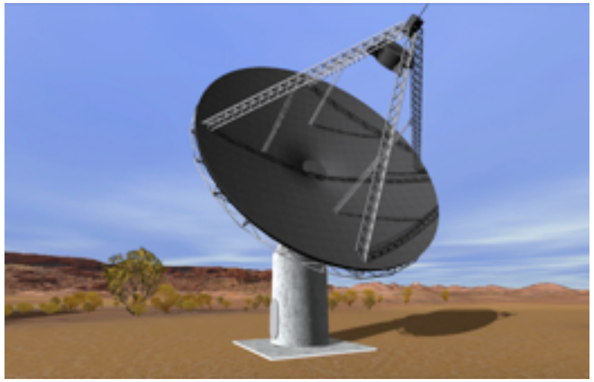




CSIRO

Current status of ASKAP

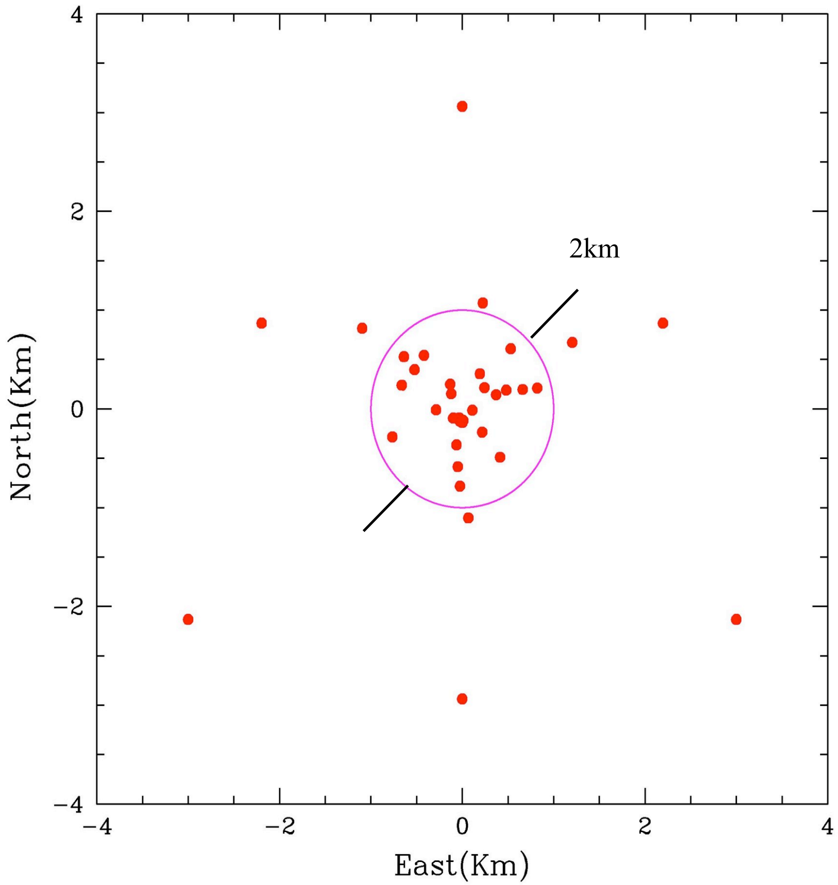
- BETA scheduled to be operational late 2010
- All subsystems going through PDR now
  - CDR Q4/2009 - Q1/2010
- Antenna contract signed
  - CETC54 from China will deliver 36 antennas
- Site acquisition still proceeding
  - Expect access Q3 2009
- Configuration defined
  - 30 antennas < 2km with excellent naturally weighted beam
  - 6 antennas out to 6km for high resolution continuum
- Science Survey Teams
  - Call for Expressions of Interests
  - Over subscription for first 5 years ~ 6
  - Down selection performed, proposals invited

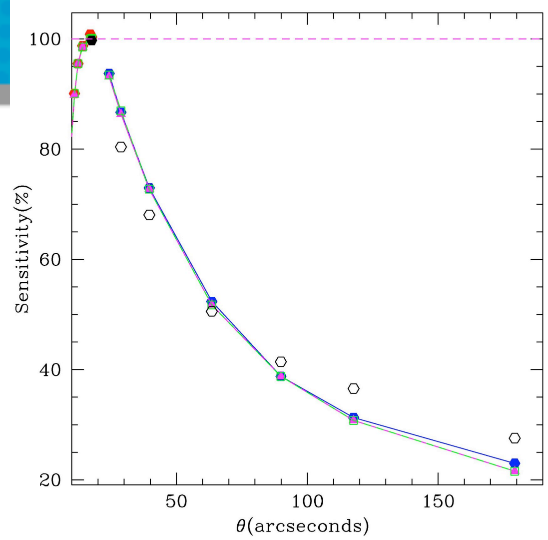


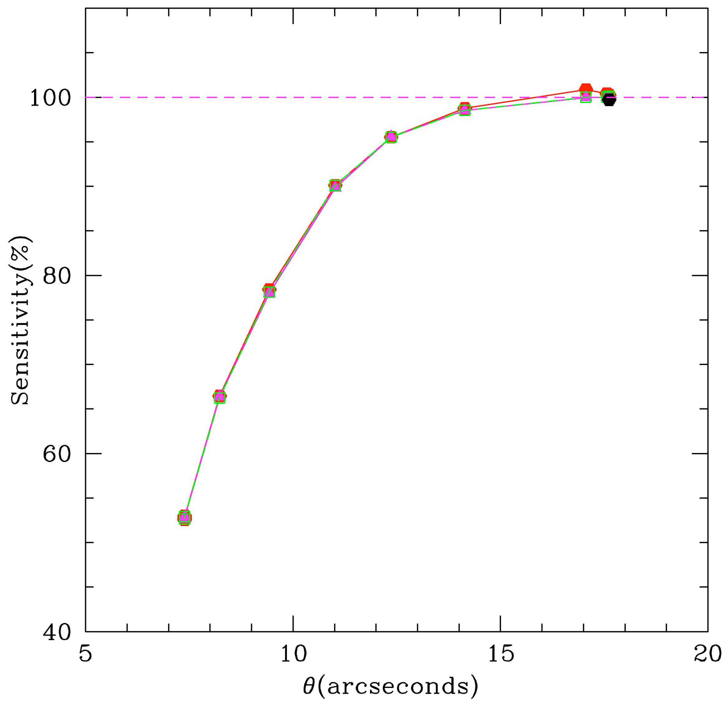
CSIRO

Configuration

2km core with 30 antennas - excellent naturally weighted PSF
6km with 36 antennas - good robust weighted PSF

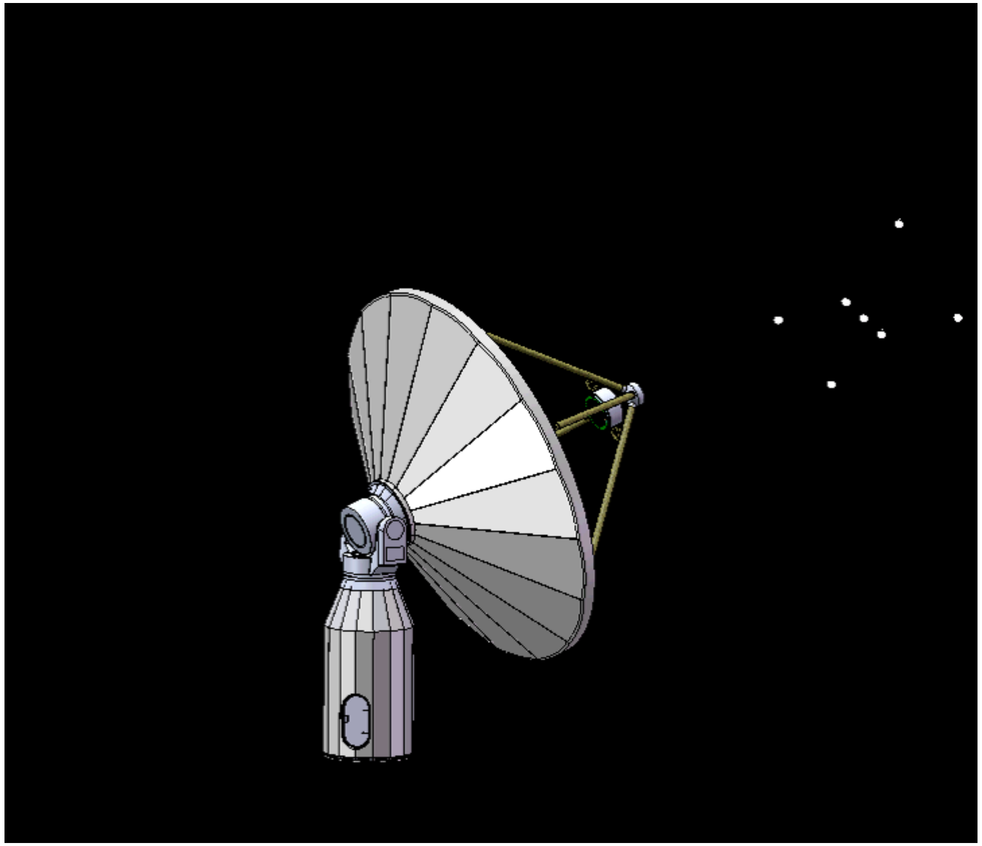


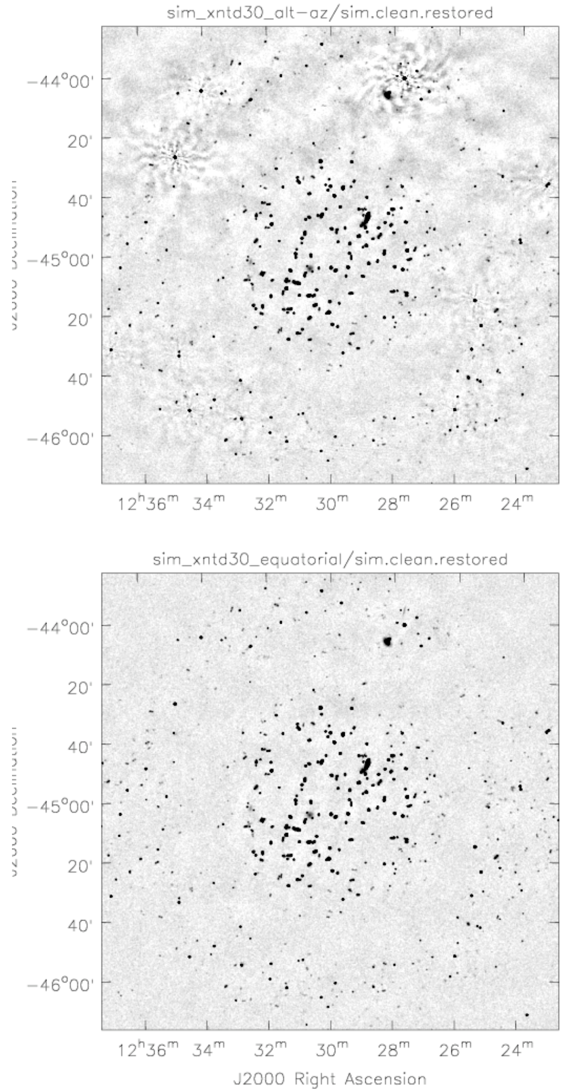




Sky mount on ASKAP antennas

- Three-axis telescope to keep antenna side-lobes (and PAF) fixed on sky
  - A triumph of software over hardware!

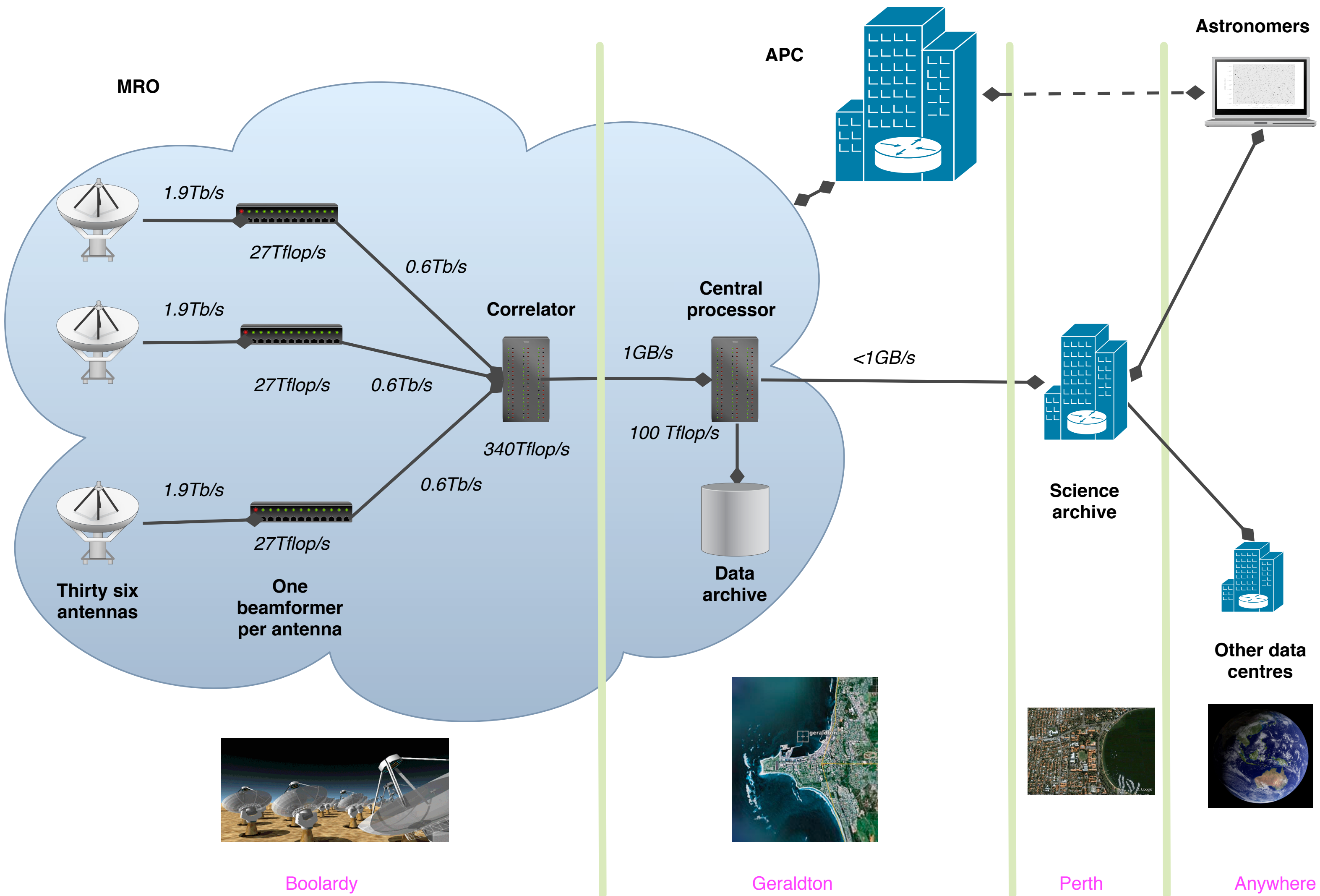




Outline

- ASKAP
- Central Processor
- Single Digital Backend

ASKAP data flow



Key ASKAP processing requirements

- Keep up with input data rate
- Process data from observing to archive with minimal human decision making
- Calibrate automatically
- Imaging
  - Fully automated processing, random field
  - Fully automated processing, Galactic plane
  - Fully automated processing, HI
- Form science oriented catalogues automatically

Calibration and imaging model for ASKAP

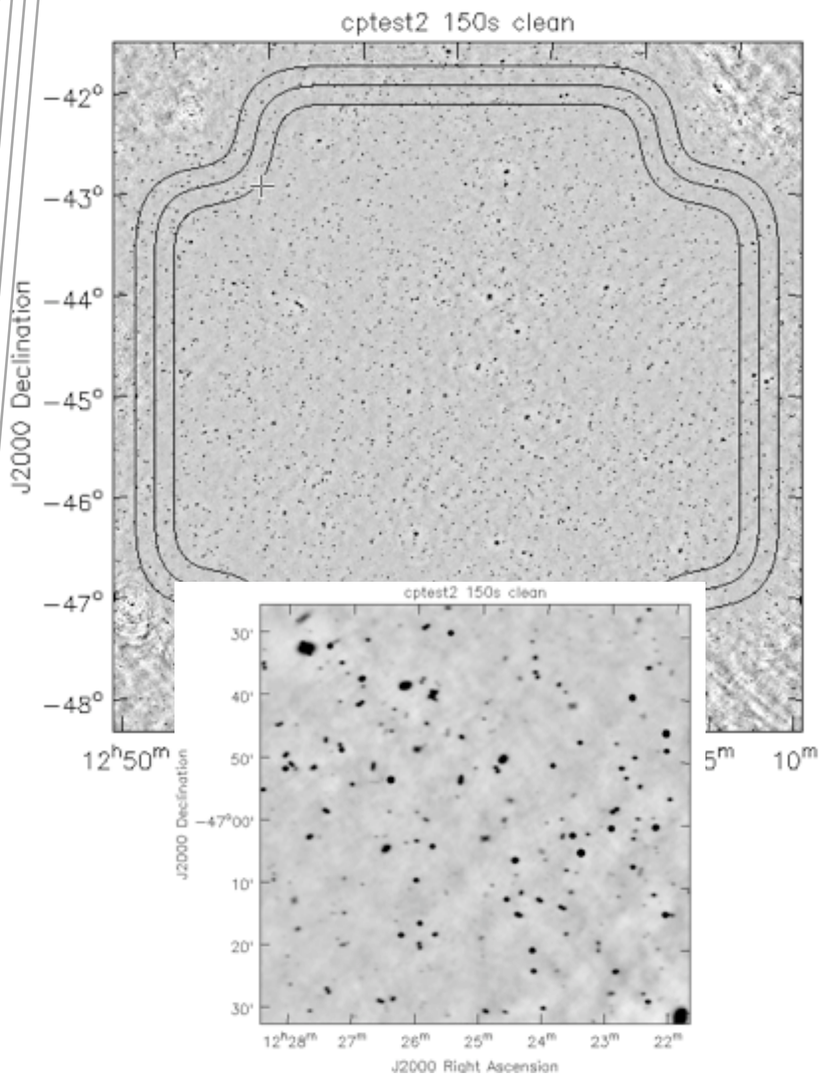
- PAF imaging similar to mosaicing
  - Use analysis from Cornwell, Holdaway, and Uson (1993)
- PAF imaging DR
  - Knowledge and application of primary beams
  - Dynamic range in synthesized beam
- Synthesized beam excellent : DR ~ 25 - 30dB
  - To reach 50dB, primary beam DR ~ 20 - 25dB
  - Quite hard to reach this level of accuracy
- Require accurate deconvolution
  - Multi-frequency, multi-scale algorithm (Urvashi Ph.D.)
- Peeling
  - Remove brightest sources by peeling
  - Estimate gain from peeling ~ 10dB additional improvement
- ASKAP model
  - Peel brightest sources (down to ~ 1Jy) using prior sky model
  - Apply accurate model of PB using illumination pattern as gridding function
  - Deconvolve using multi-frequency, multi-scale algorithm (Urvashi's talk)

Central processor status

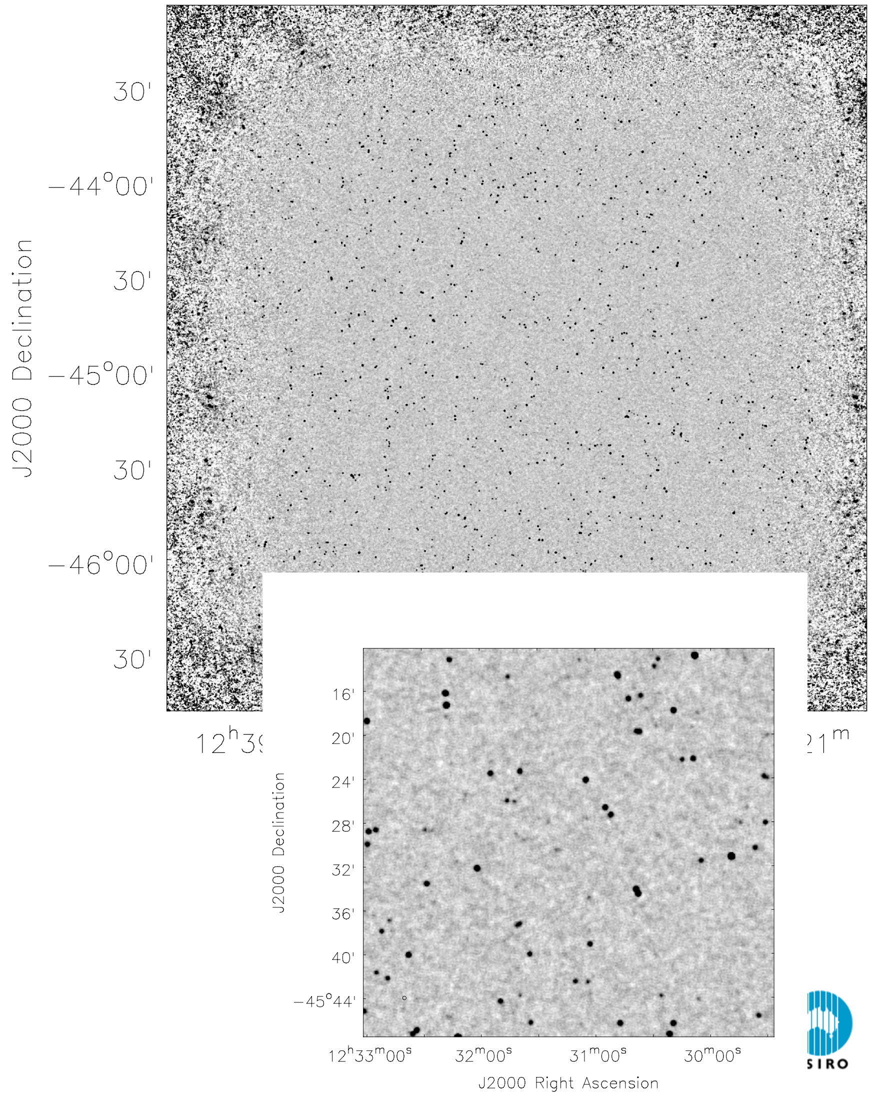
- Developed parallel imager
  - Max will talk more about design next
  - Partitioning over frequency and beams possible
  - Initial scaling demonstrated:
    - ~ 256 nodes (under IBM System S)
    - ~ 32 nodes (under MPI)
- Developed parallel source finder
  - Based on Duchamp
- Imaging algorithm advances
  - Published theory (Urvashi, Bhatnagar, Voronkov, Cornwell, IEEE in press)
  - Post hoc reweighting = preconditioning
  - Multi-frequency multi-scale (Urvashi PhD)
- Developed detailed cost model for processing
  - Vital for what-if analysis
- Investigated various hardware options
  - Excellent help from vendors
    - Intel, AMD, Blue Gene, CRAY, NEC, HP
  - GPGPU, Cell, FPGA

Simulations of radio sky observed with ASKAP

Computation limited, extended source model



Noise limited, point source model



Evaluation of Hardware

- Built a small benchmark from the core gridding / degridding algorithm.
  - About 90% of our computing requirements relate to this algorithm
- Distributed benchmark very widely
  - Along with briefing paper
- Vendors have provided benchmark numbers or machines
  - Intel (Harpertown and Nehalem CPUs)
  - AMD (Opteron 2000 series CPUs)
  - NEC (SX-8R & SX-9R)
  - SGI (SGI Altix 4700 - Itanium)
  - IBM (BlueGene/P)
  - NVIDIA (Tesla C870 GPU & GeForce GTX 260)
  - Cray (XT5 & X2)

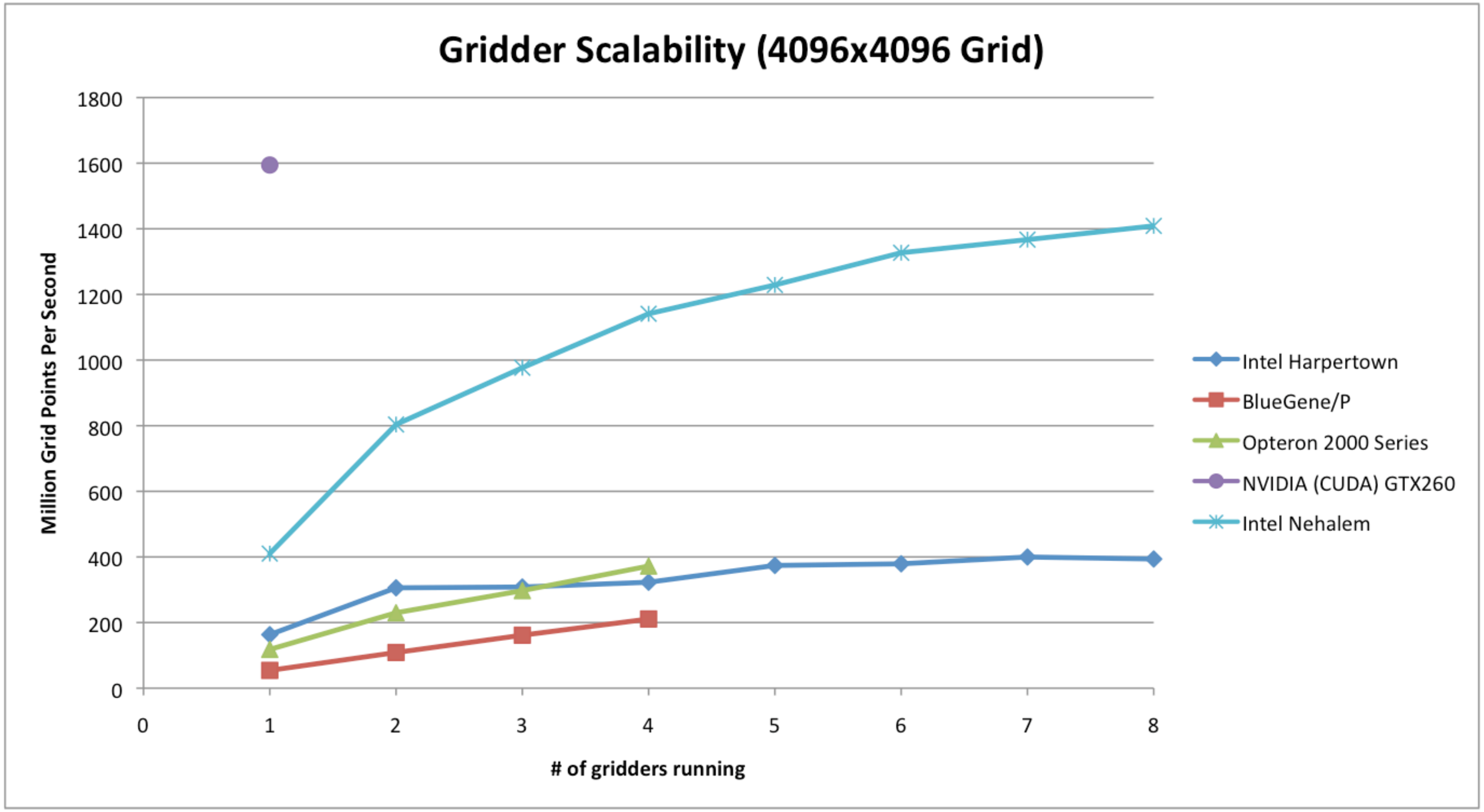
CSIRO

Gridding kernel

- Worked hard to simplify as much as possible
- Totally generic kernel
  - Adaptations at next level up
- C++ is fine - no need for FORTRAN
- Code supports pointers, BLAS, or CASA::Vector
  - Most efficient to use pointers or BLAS

```
#if defined ( ASKAP_GRID_WITH_POINTERS ) || defined ( ASKAP_GRID_WITH_BLAS )
        for (int suppv = -support; suppv < +support; suppv++) {
                int voff = suppv + support;
                int uoff = -support + support;
                casa::Complex *wtPtr = &convFunc(uoff, voff);
                casa::Complex *gridPtr = &(grid(iu - support, iv + suppv));
#ifdef ASKAP_GRID_WITH_BLAS
                cblas_caxpy(2*support+1, &cVis, wtPtr, 1, gridPtr, 1);
#else
                for (int suppu = -support; suppu < +support; suppu++) {
                        (*gridPtr) += cVis * (*wtPtr);
                        wtPtr += 1;
                        gridPtr++;
                }
#endif
#endif
```

Failure of scaling on multi-core



Other numbers are confidential

Special processors

- Special processors for convolutional resampling:
- Co processors
  - FPGA
  - Cell processors
  - GPGPUs

- Field Programmable Gate Arrays (FPGA)
  - Contract awarded to Cray/DRC Computer to prototype an FPGA for convolutional resampling.
  - Performance appeared to be promising for small grid sizes. Relative to CPU ~50x speedup.
  - Limited memory makes large grid sizes challenging.
  - The prototype took far too long to develop and requires very specialised skill set.

Special processors

- Cell Processor
  - The same processor as in the PS3 is sold by IBM as a QS22 blade.
  - Difficult programming model.
  - Very small (256Kb) local memory is problematic for our imaging algorithms.
  - Buffered processing necessary (Anna Varbanescu)
- GPGPU
  - General purpose GPU available from NVIDIA and AMD.
  - Buffer processing necessary
  - We have ported our gridding benchmark with some promising results.
  - Software development effort is larger than with a regular CPU and may cancel out any cost savings.
  - Lack of flexibility is a concern
  - Programming model liable to change
    - CUDA -> OpenCL

Performance measurement

- Metric for price/performance:
  - Price per million grid points per second
    - i.e. how much it costs to acquire a computer to perform at a certain level.
    - Typically ~ few dollars
- Metric power/performance:
  - Power per million grid points per second
    - i.e. how much power is required to perform at a certain level.
    - Typically ~ tens mW
- As of early 2009:
  - Intel Nehalem, AMD Opteron and BlueGene/P offer similar acquisition price/performance, BlueGene/P ahead in power efficiency
  - Vector machines perform well but price/performance is poor
  - NVIDIA Tesla offers best price/performance but factoring in extra development time (and hence cost) makes its value questionable.

Model for synthesis data processing costs

- Key result from investigations over last two years
- Under continual refinement
- Key parameters
  - Convolution support
  - Cost per million points per sec
  - Baseline length
- Gridding only
  - Calibration, deconvolution, and source finding not yet included

Array Parameters	Continuum	Slow transients	Fast Transients	Spectral Line
Number of antennas	36			
Max baseline length [km]	6			
Number of baselines	666			
Number of beams (feeds)	32			
Frequency channels	16384	16384	[illegible]	[illegible]
Number of polarizations	4	4	[illegible]	[illegible]
Frequency channels required	256	256	[illegible]	[illegible]
Data sizes and rates				
Bits per complex sample, weight	[illegible]			
Raw visibility frame [Gbytes]	11.71	11.71	[illegible]	11.71
Number of polarizations	4	2	[illegible]	2
Integration time [sec]	5	5	[illegible]	5
Data rate [Gb/s]	***18.73***	[illegible]	[illegible]	***18.73***
Averaged visibility frame [Gbytes]	*0.18*	[illegible]	*0.09*	*5.85*
Averaged data rate [Gbyte/s]	*0.037*	[illegible]	*91.461*	*1.171*
Averaged data rate [Tbyte/h]	*0.13*	[illegible]	*321.54*	*4.12*
Averaged data rate [Pbyte/y]	*1.10*	[illegible]	*2750.70*	*35.21*
Observation time [hrs]	12	[illegible]	0	12
Averaged visibility data set [Tbytes]	*1.54*	[illegible]	*0.00*	*49.39*
Computing costs				
Support (1D)	[illegible]	15	15	60
Number visibility samples/integration	[illegible]	*10911744*	*10911744*	*698351616*
Points to be gridded	[illegible]	*2455142400*	*2455142400*	*2.51407E+12*
Time per grid point [ns]	5			
Time to grid one integration [s]	[illegible]	*12.3*	*12.3*	*12570.3*
Number of griddings or degrid[illegible]	[illegible]	2	1	3
Number of cores needed	[illegible]	*5*	*12276*	*7542*
Million points per second	[illegible]	*982*	*2455142*	*1508439*
Cost ($) per million po[illegible]	5			
Cost M$ (2008)	***7.86***	***0.00***	***12.28***	***7.54***
Image sizes				
FOV [sq deg[illegible]	30			
Number of [illegible]	4	2	2	2
Image [illegible]	*12288*	*12288*	*12288*	*12288*
Total [illegible]	*576*	*288*	*288*	*18432*
Nu[illegible] memory	*36*	*6*	*6*	*6*
Req[illegible] size [Tbytes]	***20.7***	***1.7***	***1.7***	***110.6***
Memory [illegible] [Gbytes]	*2.6*	*351.9*	*0.1*	*14.7*
Fields per [illegible]	1000	0	0	1000
Survey size [Tb]	*563*	*0*	*0*	*18000*

These numbers are indicative only and don't reflect actual configuration or decisions.

BETA and ASKAP computing needs

- BETA hardware requirements:
  - 3-6 TFlop/s
    - 256-512 cores (as of late 2008 / early 2009)
  - 1-2 TB memory
  - Good memory bandwidth (>15 GB/s per socket)
  - 50 TB persistent storage (1 GB/s I/O rate)
  - Modest network interconnect
    - Single 1GbE for compute nodes
    - Single 10GbE for the ingest and output nodes
- ASKAP
  - 100 TFlop/s
    - ~8000 cores (as of late 2008 / early 2009)
    - ~10000 if we assume a more realistic 80% efficiency
  - 16-150 TB memory (depending on processing model)
  - Good memory bandwidth (>15 GB/s per socket)
  - 1 PB persistent storage (8-10 GB/s I/O rate)
  - Modest network interconnect
    - 1GbE for compute nodes
    - 2-4 x 10GbE for the ingest and output nodes

CSIRO

Outline

- ASKAP
- Central Processor
- Single Digital Backend

Single Digital Backend

- Experimental program to investigate replacement of custom FPGA hardware with computers built by someone else.
  - Helpful for ASKAP (50 racks of custom hardware)
    - Not on critical path
  - Vital for SKA (~1000 racks???)
- Move digital processing algorithms onto floating point units
  - More likely to be customized chips
- Take advantage of:
  - Computer packaging, power, cooling
  - Reliability engineering
  - Highly flexible network
  - High level programming tools
- Issued call for Expressions Of Interest
  - On AusTender
  - Closed 23 March 2009
  - Lots of interest from major vendors

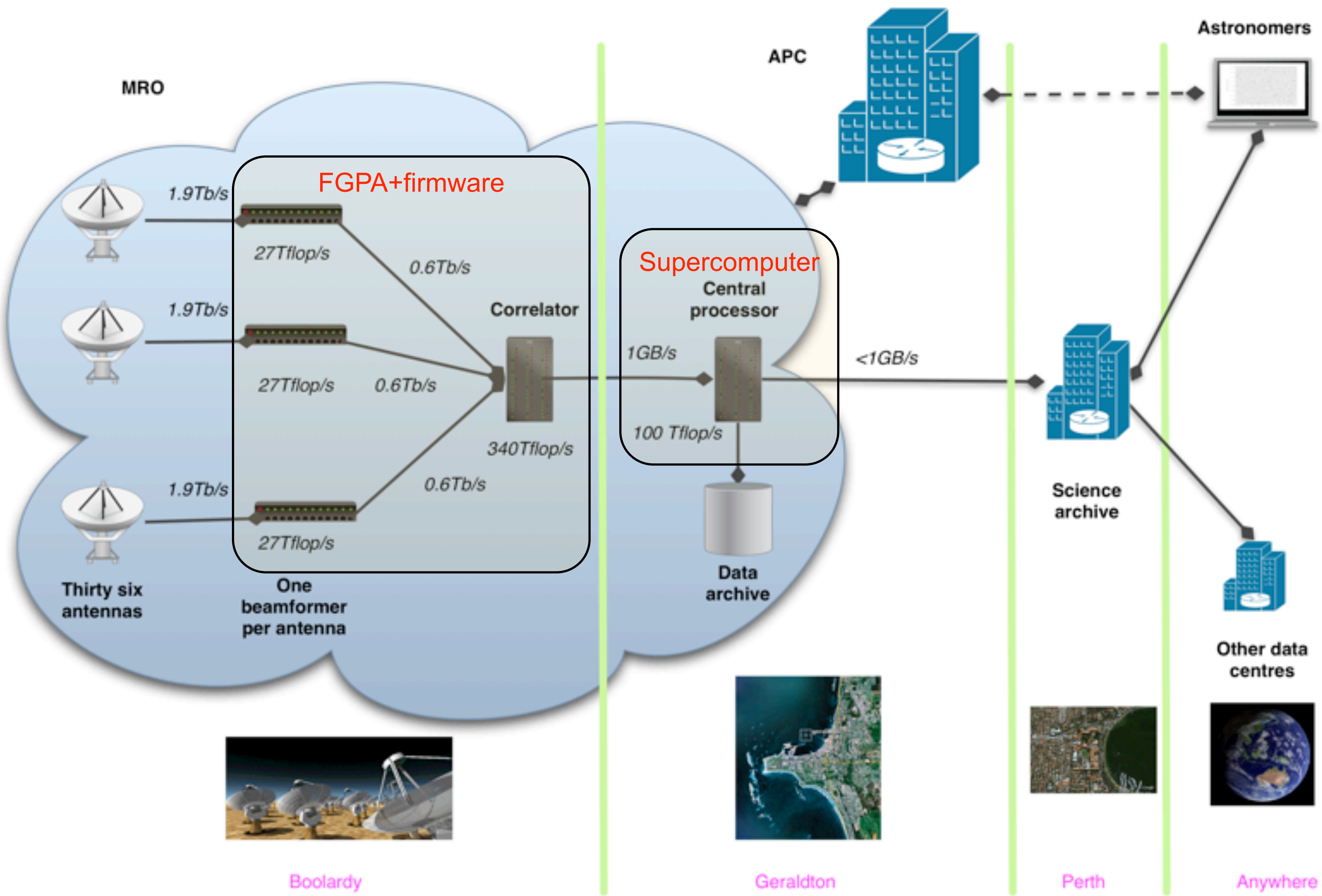
Necessity of flexibility

- Disruptive ideas
    - RFI detection and removal upstream of correlator
    - New approaches to fast transient detection
    - Lunar Cerenkov experiments
    - MOFF correlator
- Efficiency and extensibility are both valued but not aligned
- Computing hardware choices
    - Highly efficient e.g. FPGA + firmware + custom hardware
    - Highly extensible e.g. general purpose supercomputer + C++ + MPI

ASKAP SDB beam forming and correlation load

- Computing load
  - 2PFlop/s
  - Enormous input data rate ~ 72Tb/s
  - Moderate output data rate ~ 5Gb/s
  - Lots of Complex Multiply ACcumulates (CMACs)
  - Moderate length FFTs
  - Heavily streamed - not much memory needed
  - Two major data reshuffles
    - Beamforming - elements to elements for each antenna
    - Correlation - antennas to antennas for each beam
- Derived requirements
  - High input capability
  - Low computational intensity
  - Interconnect flexibility
  - High interconnection bandwidth
  - Moderate memory
  - Power efficiency

ASKAP data flow, processing, and storage



Processing steps in current scheme

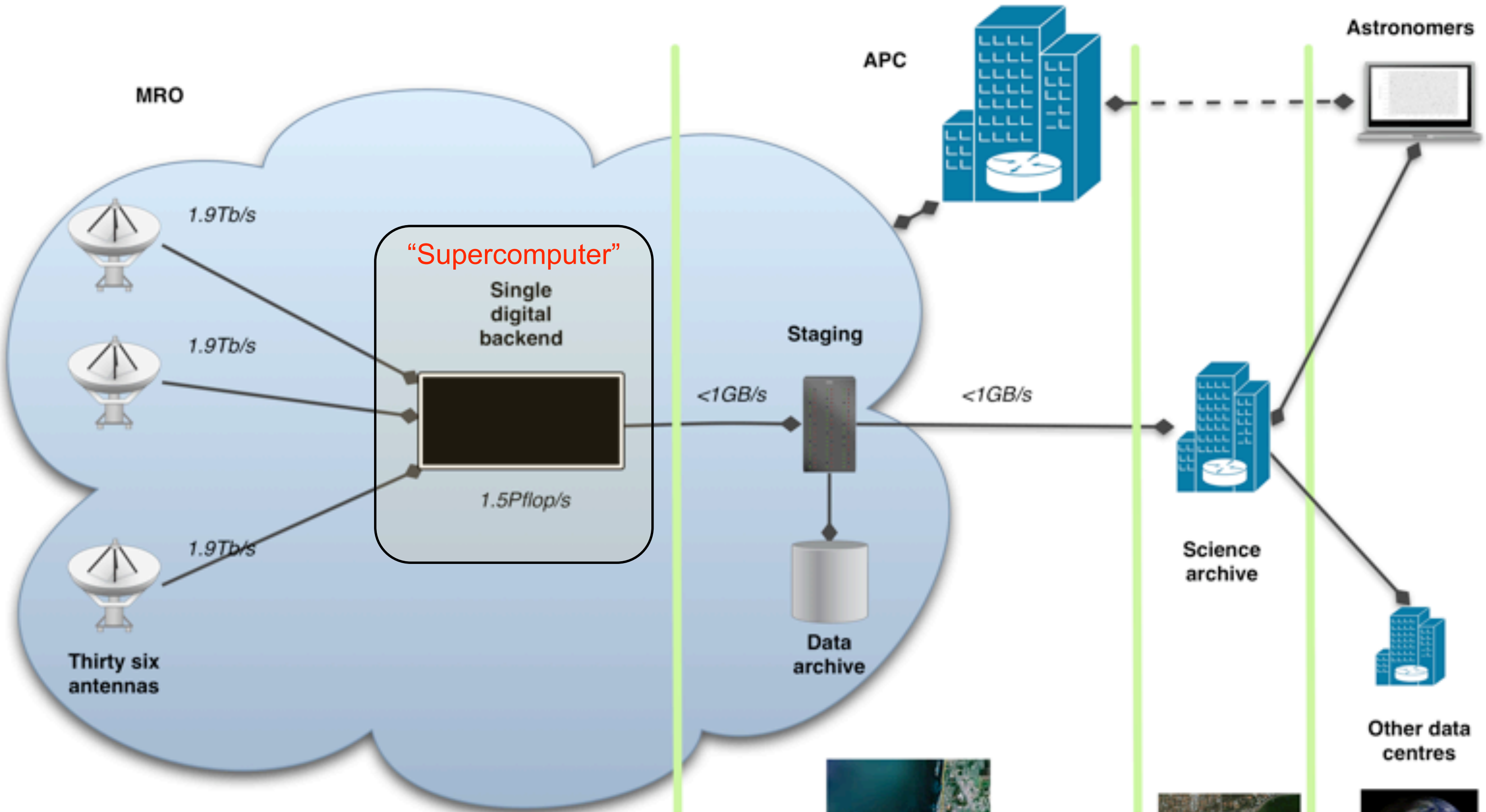
FGPA+firmware

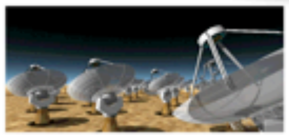
- Digital sampling
  - Produces 256 coarse frequency channels per element per antenna
  - 192 x 256 x 36 complex samples every 1us
- Beam formation
  - Linear combinations of element sampled voltages per antenna
  - 32 x 256 x 36 complex samples every 1us
- Correlation
  - Cross correlations of all antennas with all antennas by beam by frequency channel
  - Produces 666 x 32 x 16384 separate correlations per 5s

Supercomputer

- Imaging
  - Gridding all visibility samples by frequency channels
  - Produces 4096 x 4096 x 16384 image in ~ 8 hours
- Catalog extraction
  - 2D and 3D source finding

ASKAP Single Digital Backend






Boolardy Geraldton Perth

Processing steps with SDB

- Digital sampling — FGPA+firmware
  - Produces 256 coarse frequency channels per element per antenna
  - 192 x 256 x 36 complex samples every 1us
- Beam formation — Supercomputer
  - Linear combinations of element sampled voltages per antenna
  - 32 x 256 x 36 complex samples every 1us
- Correlation
  - Cross correlations of all antennas with all antennas by beam by frequency channel
  - Produces 666 x 32 x 16384 separate correlations per 5s
- Imaging
  - Gridding all visibility samples by frequency channels
  - Produces 4096 x 4096 x 16384 image in ~ 8 hours
- Catalog extraction
  - 2D and 3D source finding

ASKAP SDB beam forming and correlation load

- Computing load
  - 2PFlop/s
  - Enormous input data rate ~ 72Tb/s
  - Moderate output data rate ~ 5Gb/s
  - Lots of Complex Multiply ACcumulates (CMACs)
  - Moderate length FFTs
  - Heavily streamed - not much memory needed
  - Two major data reshuffles
    - Beamforming - elements to elements for each antenna
    - Correlation - antennas to antennas for each beam
- Derived requirements
  - <u>High input capability</u>
  - Low computational intensity
  - <u>Interconnect flexibility</u>
  - <u>High interconnection bandwidth</u>
  - Moderate memory
  - <u>Power efficiency</u>

Comparison

- Custom hardware
  - FPGA + custom firmware + custom packaging
  - Lower cost, power
  - High NRE (engineering time)
  - Lower reliability?
- Someone else's hardware (SEH)
  - e.g. Blue Gene + custom software
  - Higher cost, power
  - Much lower NRE
  - High reliability?

First step ~ 30 * LOFAR correlator

- Digital sampling FGPA+firmware
  - Produces 256 coarse frequency channels per element per antenna
  - 192 x 256 x 36 complex samples every 1us
- Beam formation
  - Linear combinations of element sampled voltages per antenna
  - 32 x 256 x 36 complex samples every 1us
- Correlation SDB
  - Cross correlations of all antennas with all antennas by beam by frequency channel
  - Produces 666 x 32 x 16384 separate correlations per 5s
- Imaging CP
  - Gridding all visibility samples by frequency channels
  - Produces 4096 x 4096 x 16384 image in ~ 8 hours
- Catalog extraction
  - 2D and 3D source finding

Summary

- ASKAP on course
  - 6 antenna test array in 2010
  - 36 antenna full array in 2012
- Improved understanding of calibration and imaging processing
  - Coarse level parallelization in place
  - Fine level (OpenMP) seems to be less important
  - Developed costing model
  - For BETA, will purchase ~ 256 core cluster
  - For ASKAP, options are still open
- Validation and verification of processing
  - Conducting end to end tests
  - Also reduction of ATCA mosaic data
- SDB
  - Long term project to move all digital processing to non-FPGA computer
  - Expressions of interests received and being reviewed
  - Very exciting expansion of capabilities

ATNF/ASKAP
Tim Cornwell
ASKAP Computing Project Lead

Phone: +61 2 9372 4261
Email: tim.cornwell@csiro.au
Web: www.atnf.csiro.au/people/tim.cornwell

www.csiro.au

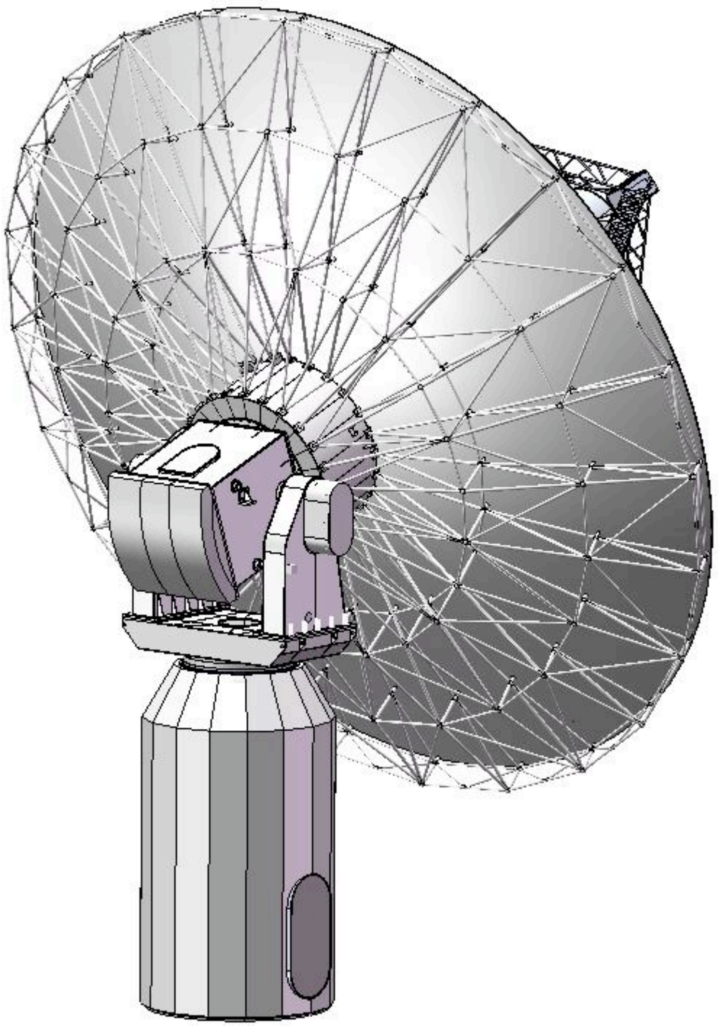
Thank you

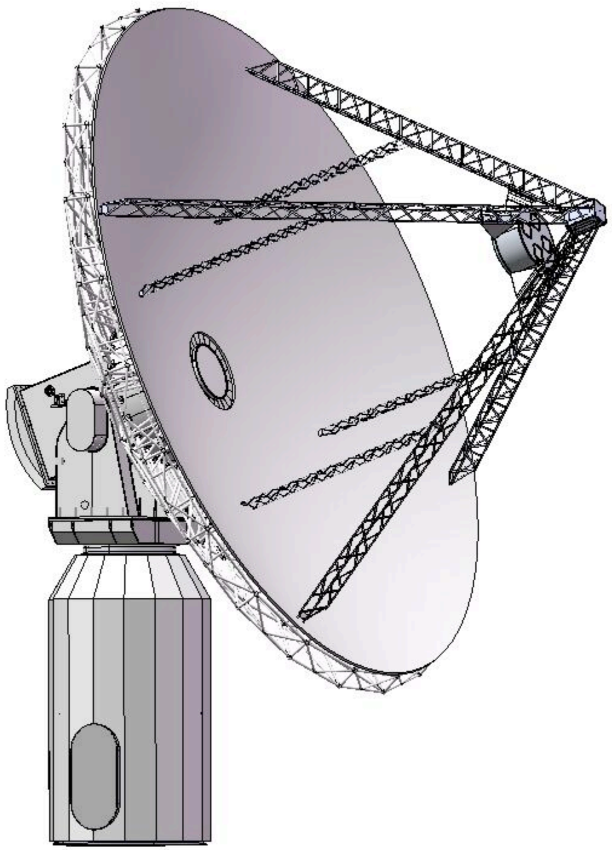
Contact Us
Phone: 1300 363 400 or +61 3 9545 2176
Email: enquiries@csiro.au Web: www.csiro.au

CSIRO

Antenna Contract

: of China Electronics Technology
C54)
ery AU$9.9M



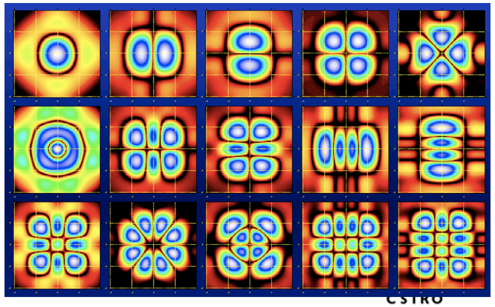


Processing models

- Streamed
  - Visibility data is processed immediately
  - Few optimization options
  - Need enough memory to store images (Over 100TB for 16384 ch)
  - Time to finish off
  - Difficult to handle faults without loosing channels
- Buffered
  - Data is stored to disk for part or all of observation
  - When enough data is ready, processing starts
  - High latency
  - Reduced memory requirement (16-32TB for 16384 ch)
  - Many choices to optimize sequence of processing and hence performance
  - Simplifies fault handling
  - Impossible if visibility rate exceeds disk I/O capability

Calibration of Phased Array Feeds

- Each PAF has 192 single pol (X,Y) elements
  - Formed into ~ 30 dual pol feeds
- Initial model is to calibrate upstream of correlator
  - Feed calibration parameters to beamformers - every 1MHz of 300MHz
- Use switched emitter(s) on antenna surface
- Measure covariance matrix of element voltages
- Solve for optimal beamformer weights every few minutes
  - Numerous criteria - max SNR, min noise, min sidelobe, prototype beam
- Takes care of two effects
  - Noise coupling between neighboring elements
  - Gain variations
- More sophisticated approach
  - Calibrate from covariance matrix and prior sky model
  - ASKAP memo by Voronkov and Cornwell
  - Not possible to solve for all elements
  - But can solve for eigenmodes



Central Processor middleware

- Need software layer for parallelization
- Three potential middleware/frameworks identified:
  - MPI
  - IBM System S
  - ICE
- MPI based imager
  - Developed as part of a collaboration between ASTRON and CSIRO
- IBM System S based imager
  - Developed as part of an evaluation of System S (InfoSphere Streams)
- Ice based imager
  - Developed as part of an evaluation of Ice
- All three of these work
- MPI imager used for testing, reduction of ATCA mosaic data

Evaluation of Middleware

- Initial imager developed in MPI
- Conducted detailed evaluation of IBM System S
  - Compelling model of stream processing - very well matched to SKA
  - Will be marketed as InfoSphere Streams
  - We cannot currently make sufficient use of all capabilities
- We have selected ICE for the front-end of the central processor
  - ICE is supported on the most major Unix platforms (Linux, Solaris, HP-UX, FreeBSD)
  - Very easy to write prototype imager with ICE
- We have selected MPI for the back-end of the central processor
  - MPI is supported on practically all HPC platforms
  - Looking at the possibility of replacing MPI with ICE for the backend