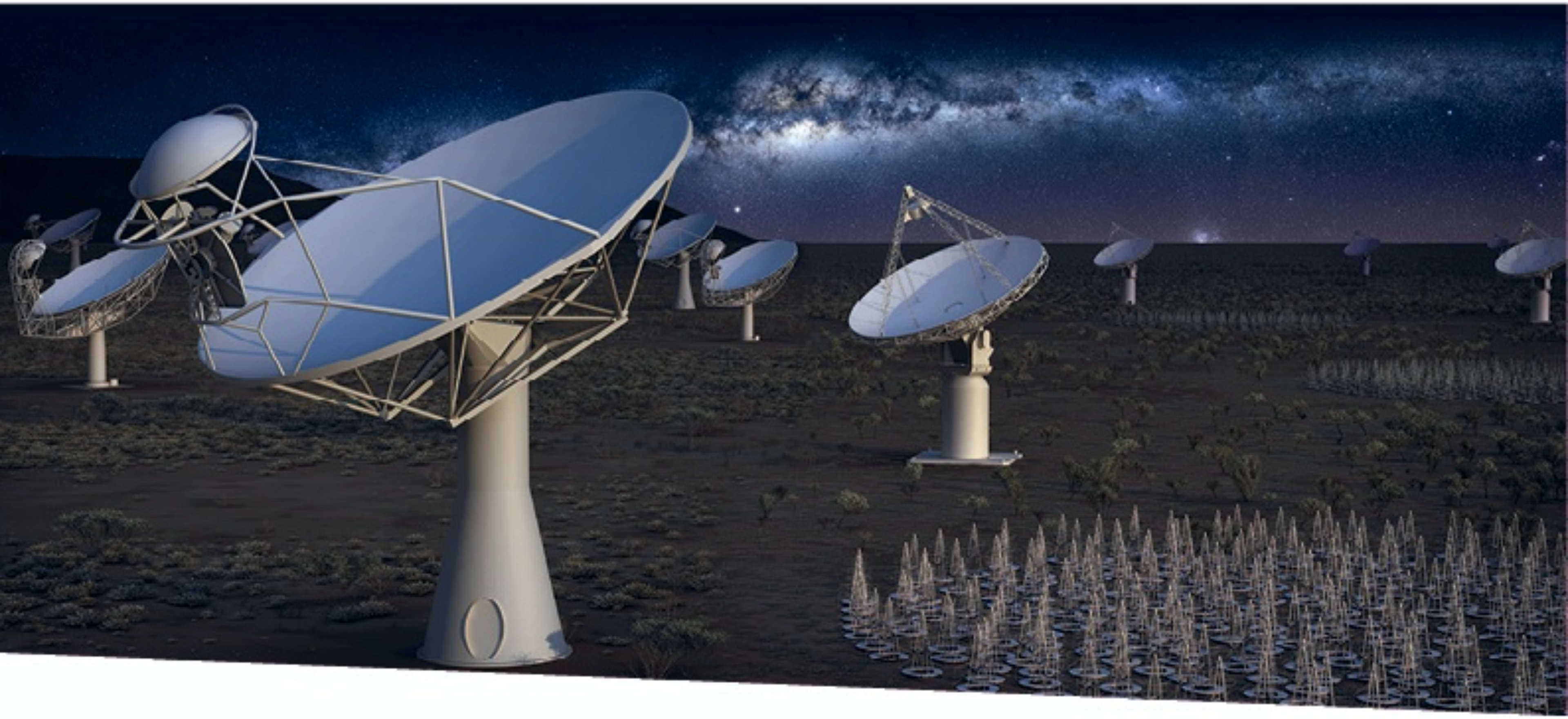
SKA Computing and Software



SQUARE KILOMETRE ARRAY

Exploring the Universe with the world's largest radio telescope

Nick Rees

18 May 2016

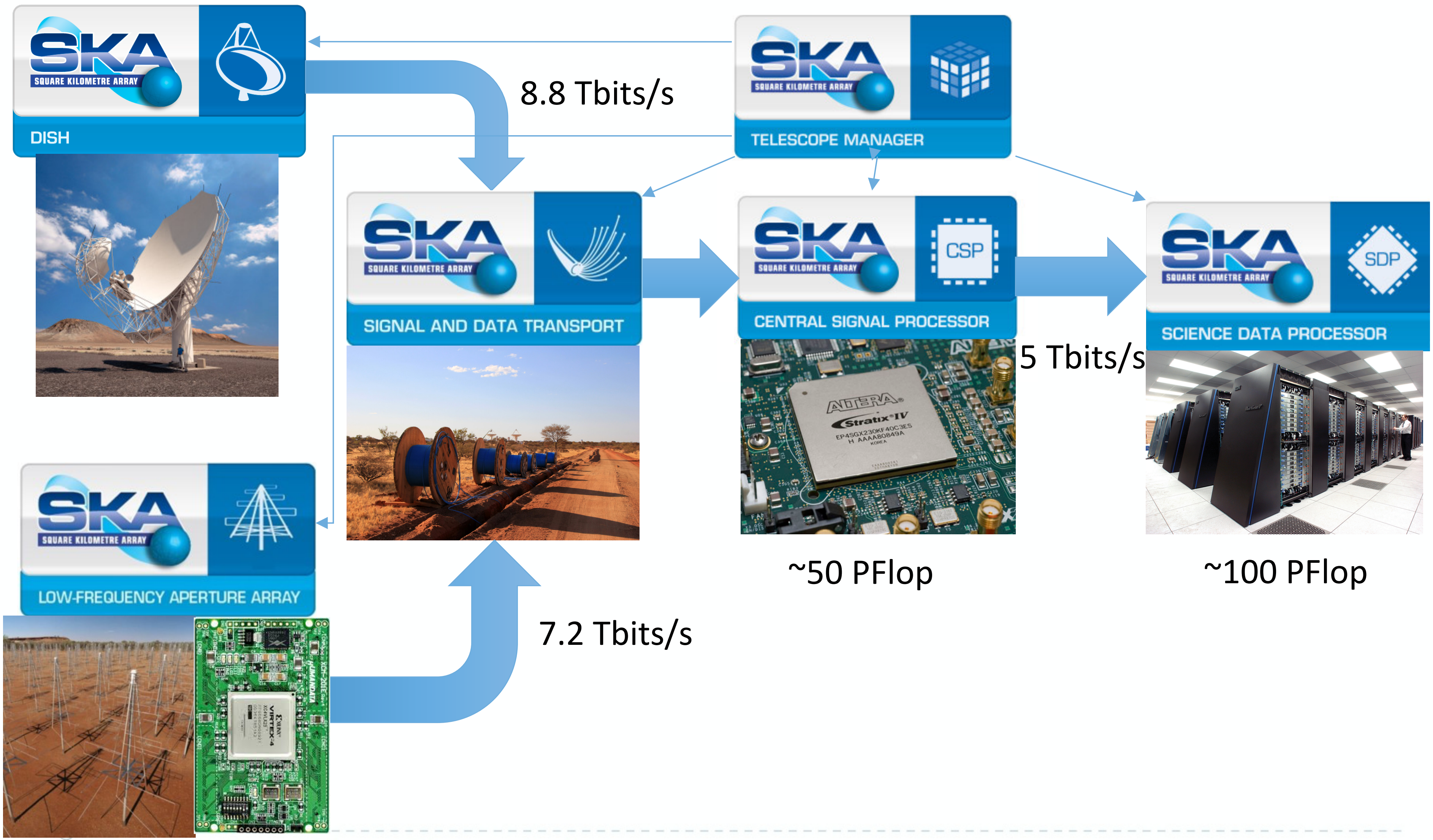
Summary

- Introduction
  - System overview
- Computing Elements of the SKA
  - Telescope Manager
  - Low Frequency Aperture Array
  - Central Signal Processor
  - Science Data Processor
- Conclusions

Introduction

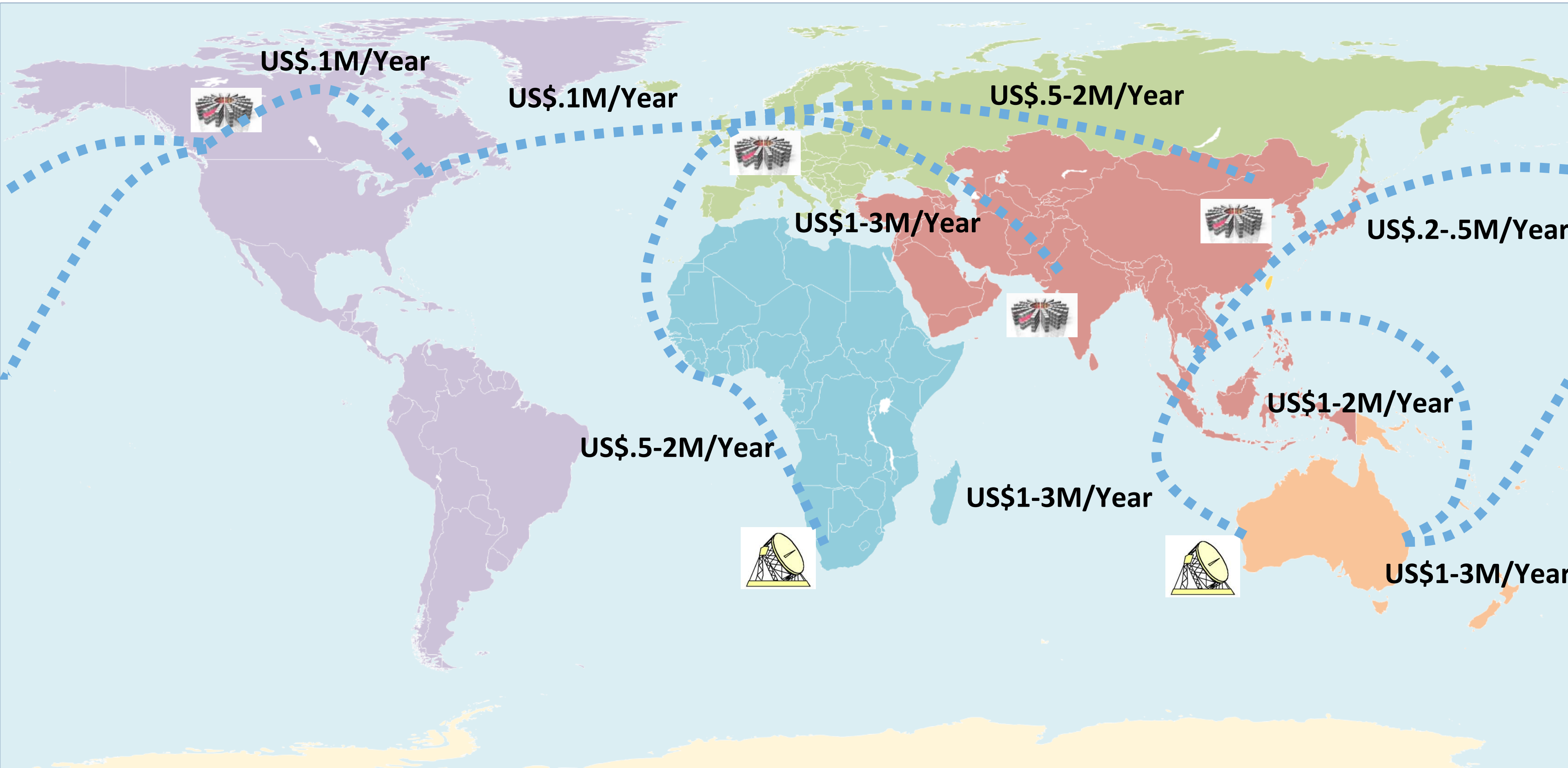
- SKA is a software telescope
  - Very flexible and potentially easy to reconfigure
  - Major software and computing challenge
- Computing challenges are huge
  - Science Data Processor needs 100 PetaFLOPS/sec of delivered processing
    - Current Top 500 supercomputer is Tianhe-2 – 50 PetaFLOPS/sec
- Software challenges are also huge
  - SDP development is baselined at ~100 FTE for 6 years.
- Talk based on current status after recent PDR's
  - Designs will evolve between now and CDR.

SKA – System Overview



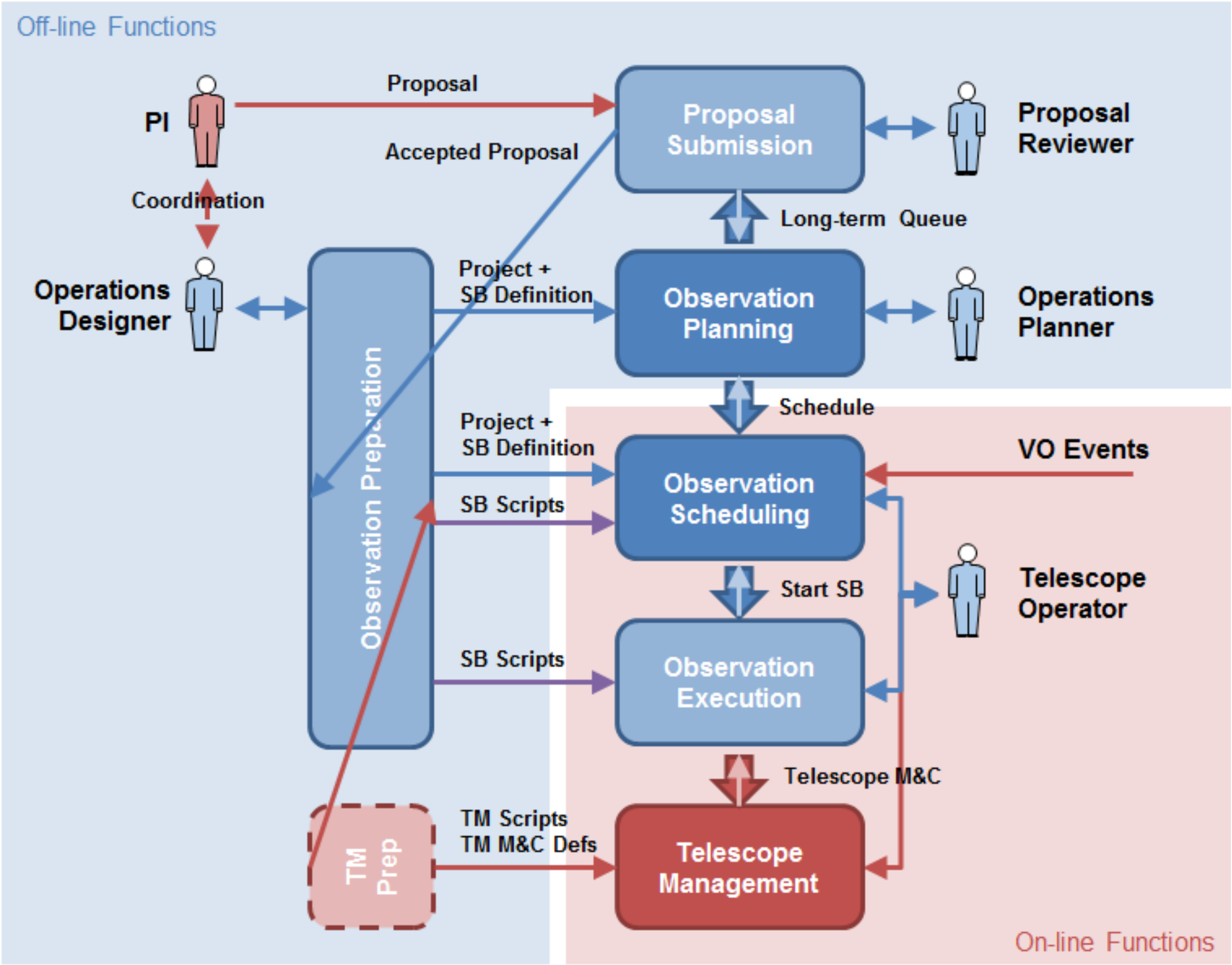
Regional Centre Network

- 10 year IRU per 100Gbit circuit 2020-2030
- Guesstimate of Regional Centres locations

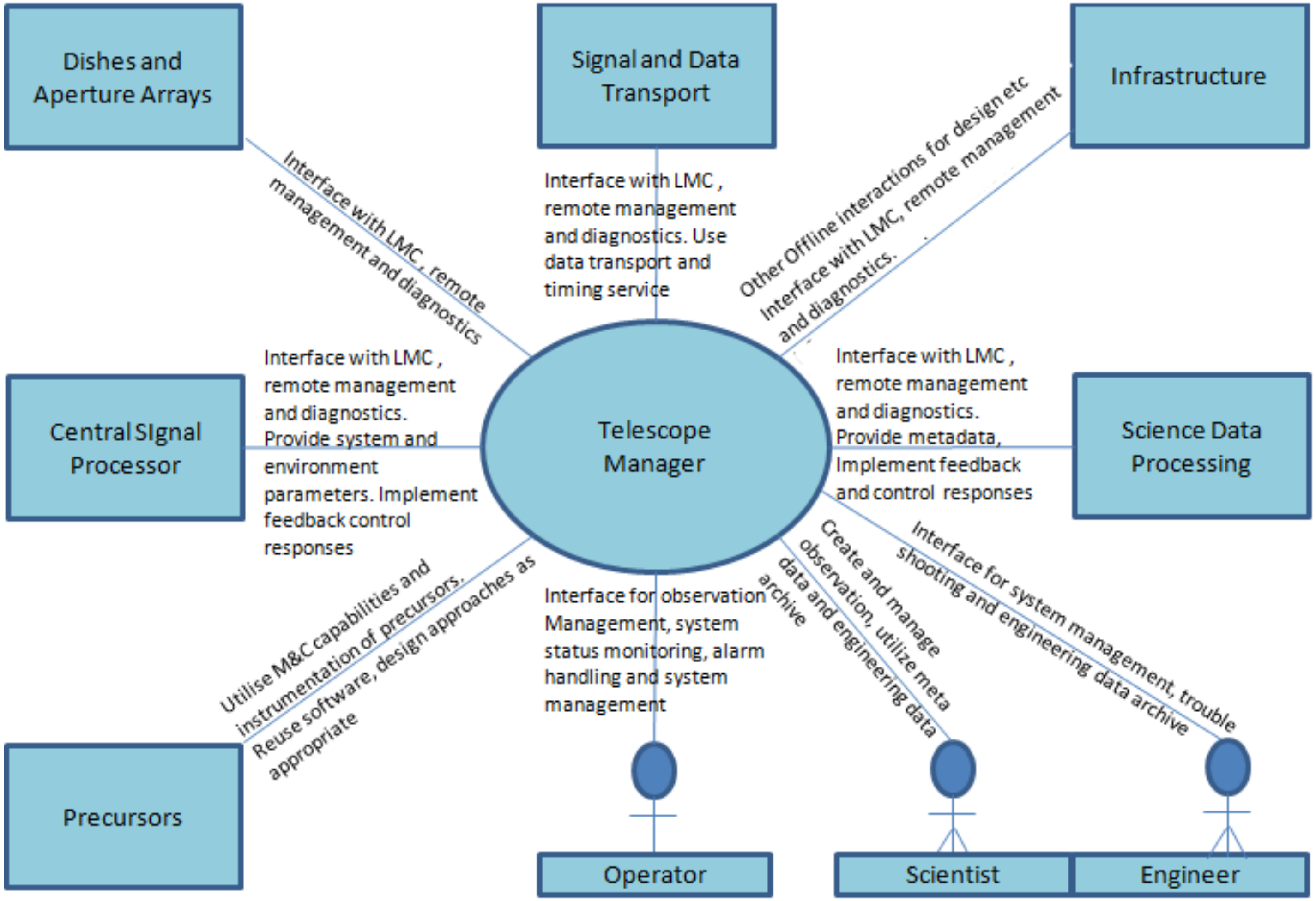


Telescope Manager

Telescope Manager Overview

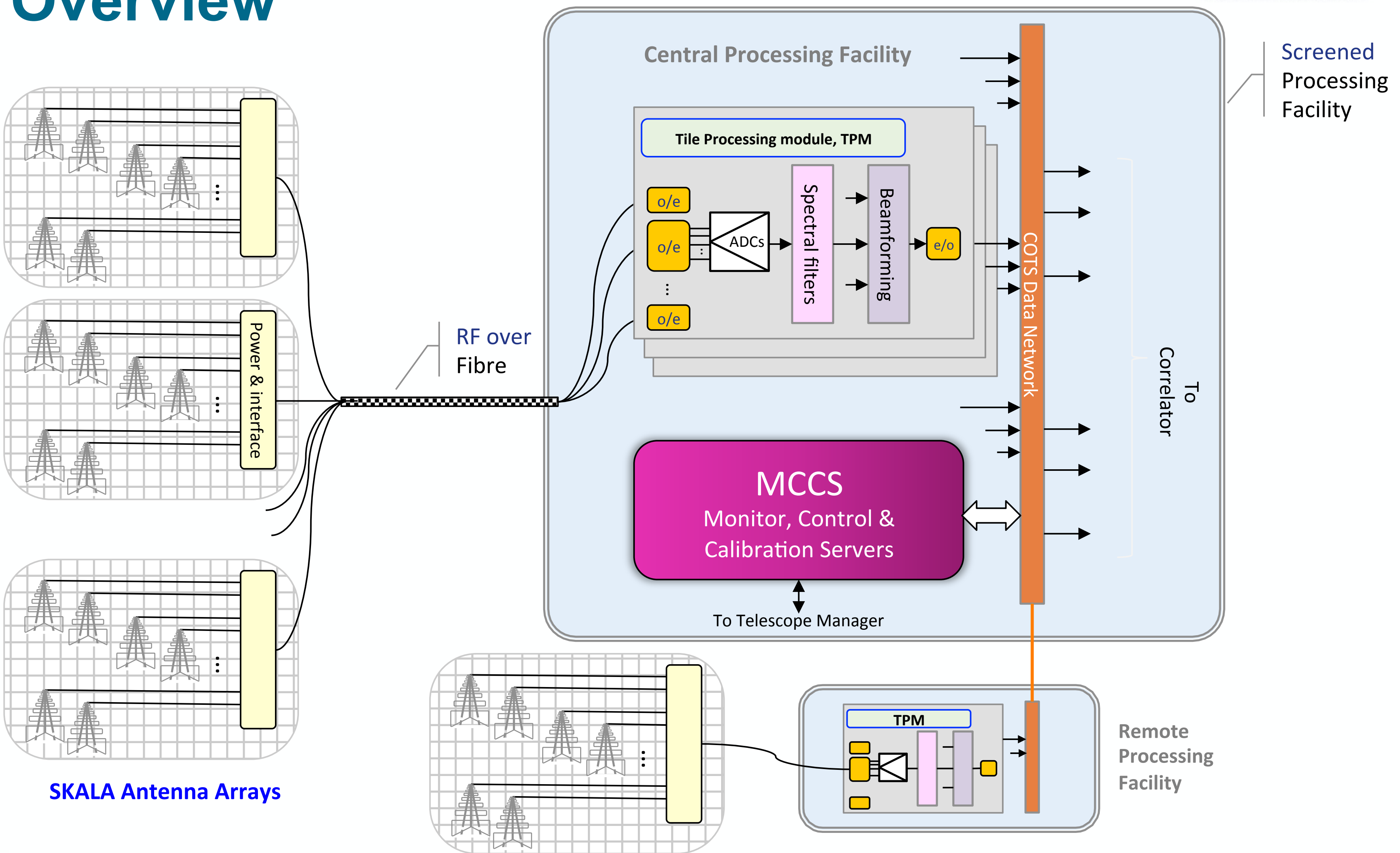


Telescope Management

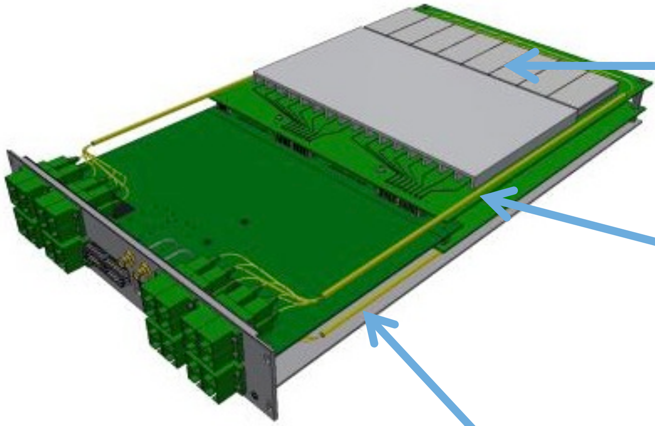


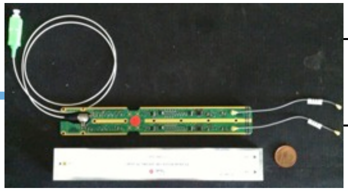
Low Frequency Aperture Array

Low Frequency Aperture Array Overview

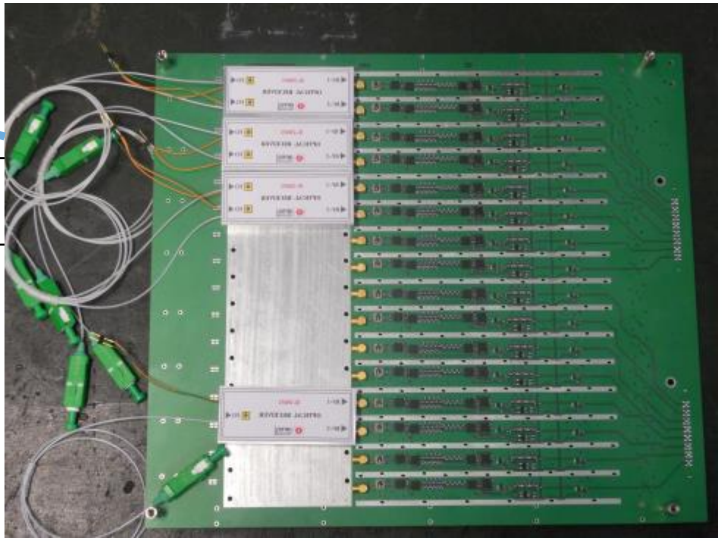


Tile Processing Modules

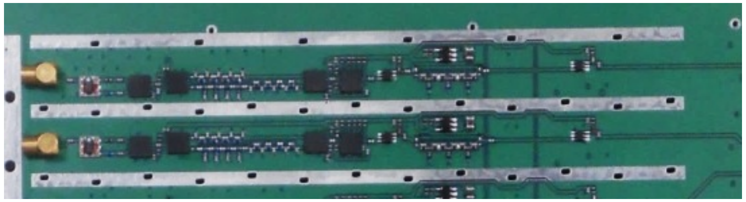


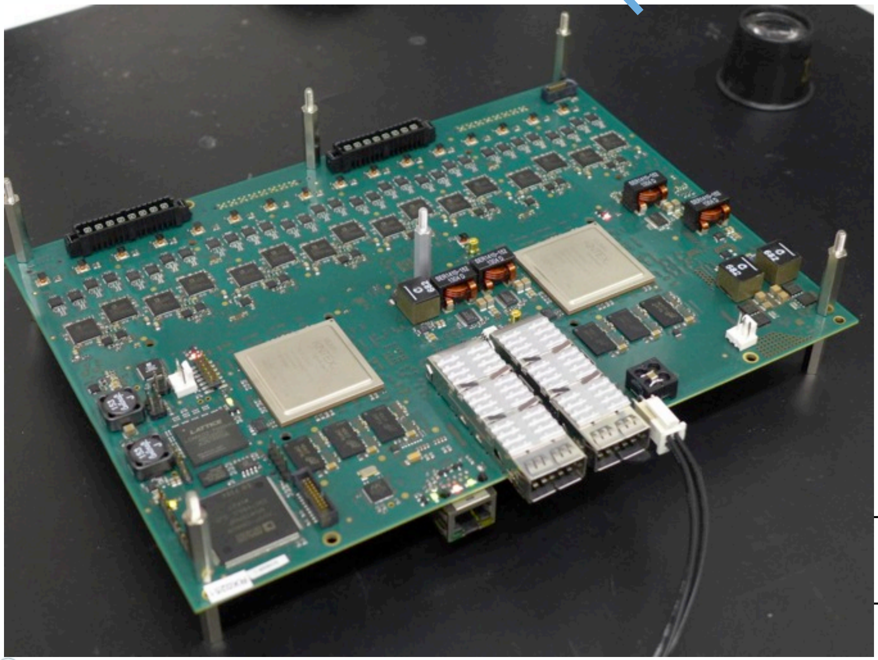


WDM optical



Analogue

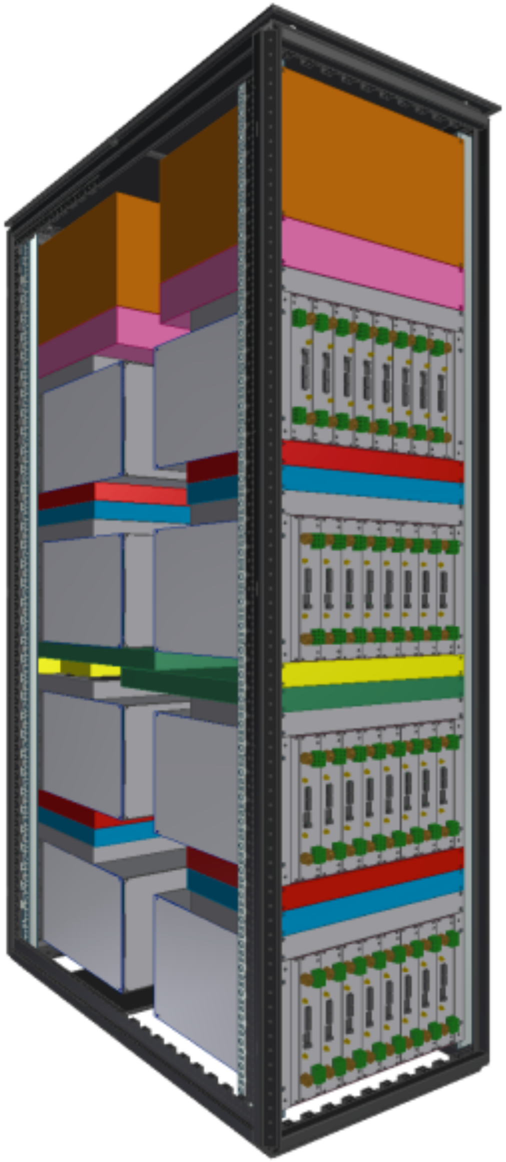




FPGA/Digital

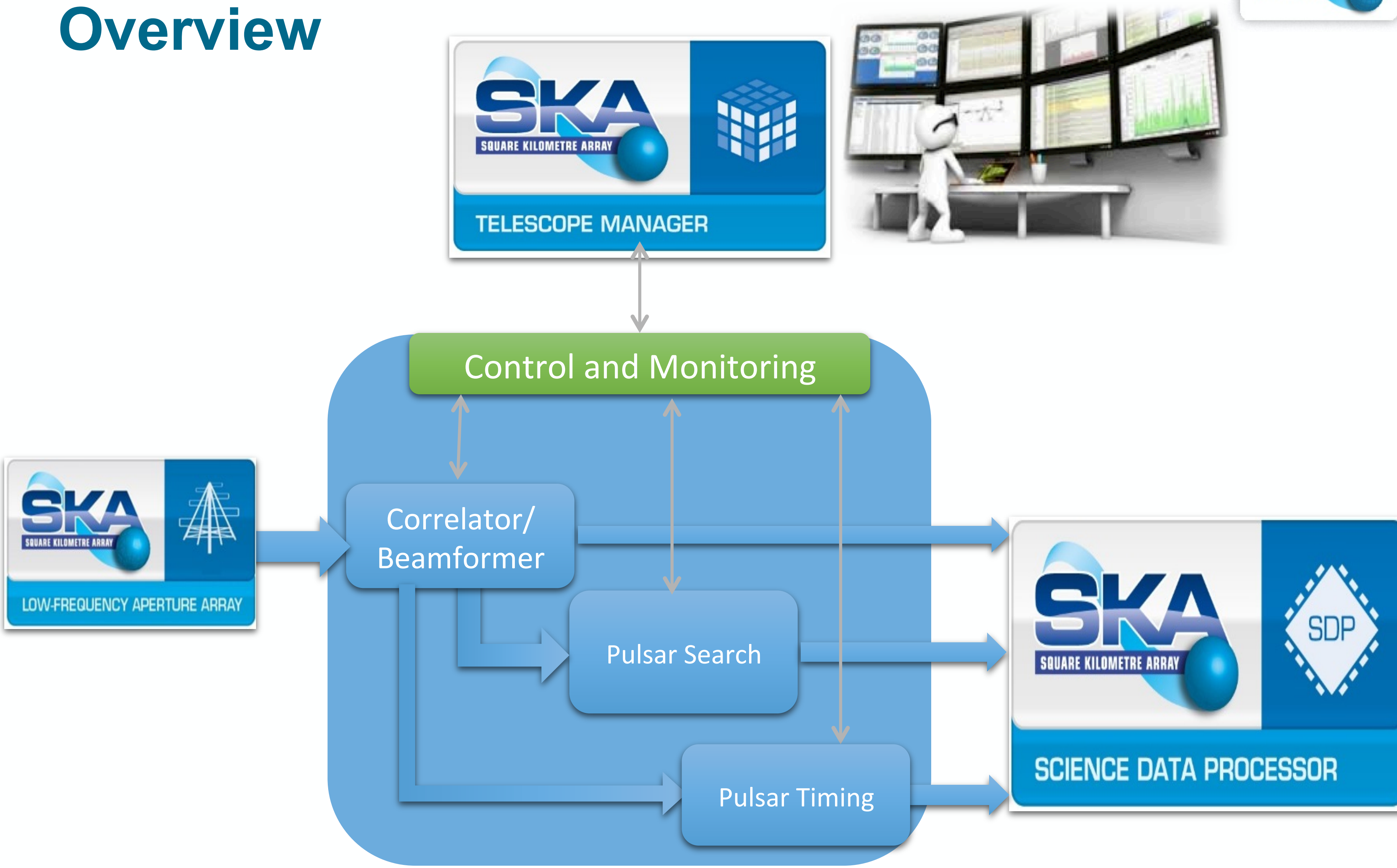
Rack arrangement

- Double sided racks
- Water cooled using building circulation
- 64 TPMs per rack, 128 in the system
- Power, Clocks and timing circulated



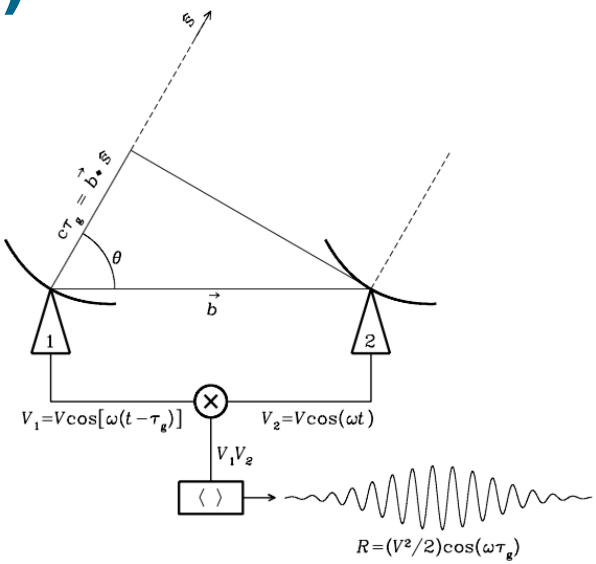
Central Signal Processor (CSP)

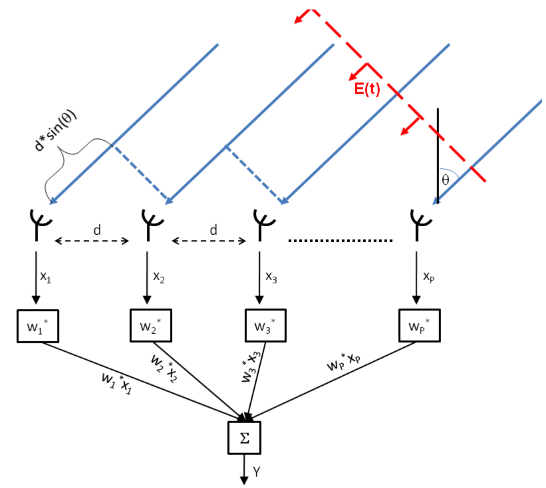
Central Signal Processor Overview



Correlator Beam Former (CBF)

- Correlator:
  - Channelise signal from every dish/aperture array station in to fine frequency channels (65k)
  - Cross-correlate all channels for every pair of dishes/stations
  - Cross-correlations ('visibilities') passed to SDP for imaging
- Beamformer:
  - Forms multiple beams within the dish/station beam
    - 1500 beams for Mid and 750 for Low
  - Passes data to Pulsar Search/Timing engines/VLBI interface
- Very large amounts of real-time processing:
  - N_{corr} ~ $B(N_{dish}.\log_2(N_{ch}) + N_{dish}^2)$ ~ PetaMAC/s
  - N_{BF} ~ $B.N_{dish}.N_{beam}$ ~ few PetaMAC/s
- Based on custom FPGA processing platforms



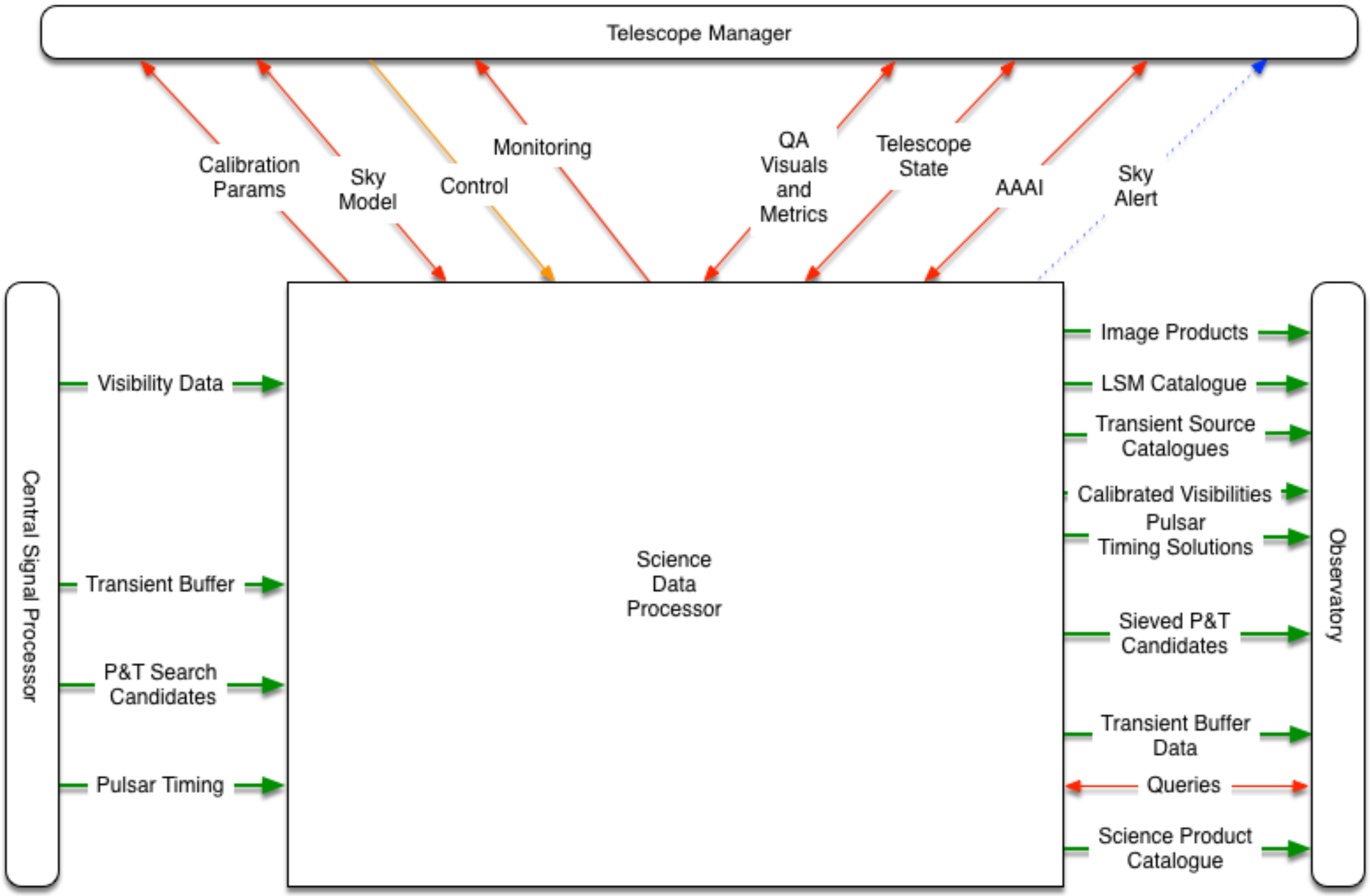


Pulsar Search

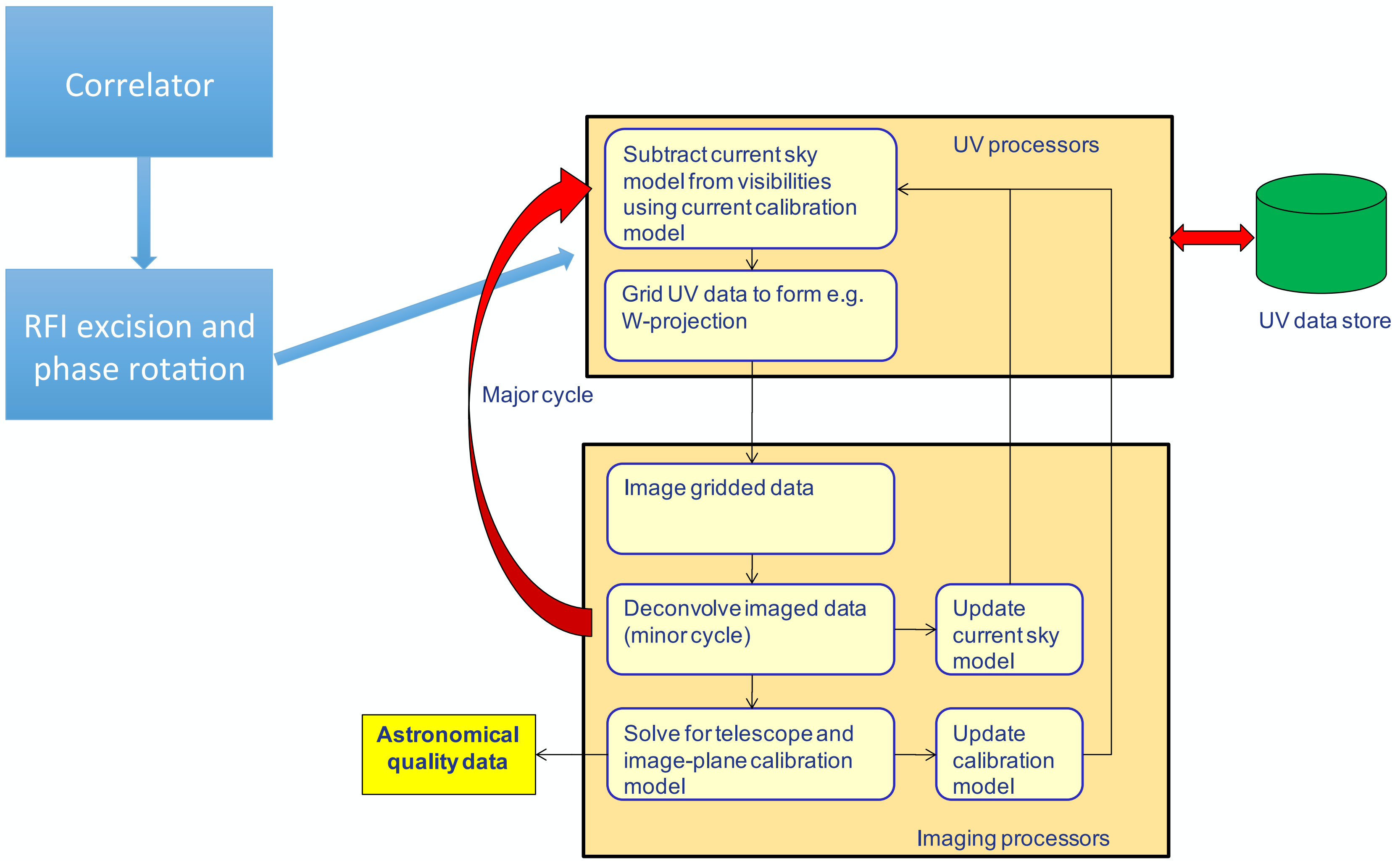
- General processing pipeline requires ~50 PFLOPS/sec.
- Baselined heterogeneous design to achieve best combination of hardware & software firmware.
- Two beams per compute node in current design.
- 250 server nodes in Australia and 750 in South Africa
- Dual redundant 10 & 1 gig networks.
- Each Node (1000 in total):
  - Low Power CPUs
  - GPUs
  - FPGA boards
  - 10 Gig inputs
  - > 1 Tbyte RAM &/or SSDs

Science Data Processor

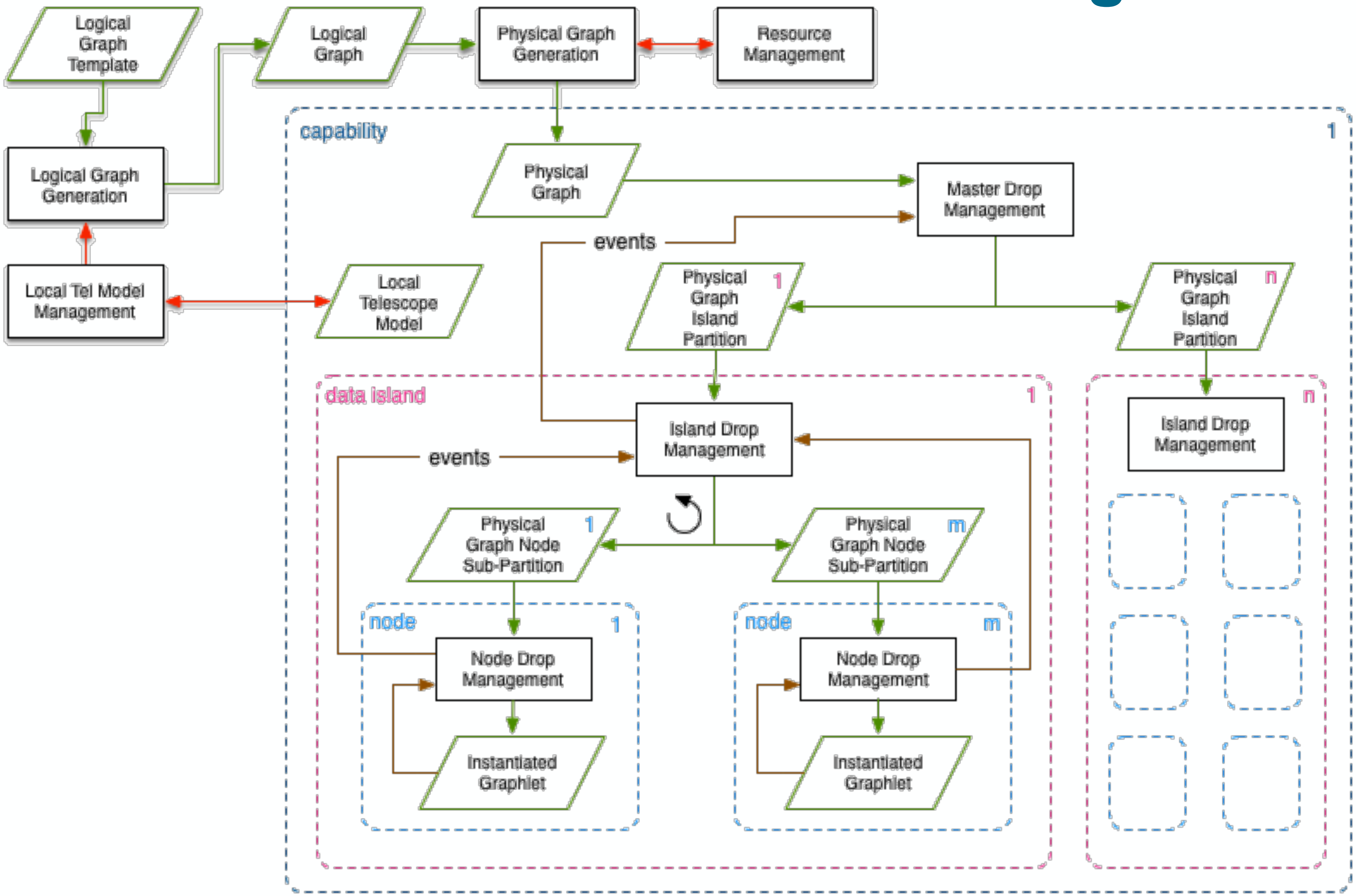
Science Data Processor Overview

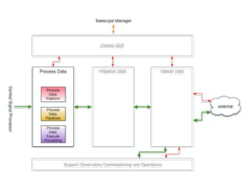


Imaging Processing Model



SDP Data Flow Approach: Next Generation Data Science Engine?





Computing Limitations

- Arithmetic Intensity ρ = Total FLOPS/Total DRAM Bytes
- The principal algorithms required by SDP (gridding and FFT) are typically $\rho \approx 0.5$
- Typical accelerators have an $\rho \approx$ 5-7
  - For example, NVidia Pascal GPU architecture has:
    - Memory bandwidth ≈ 720 GB/sec
    - Floating point bandwidth ≈ 5,000 GFLOPS/sec
    - $\rho = 5000/720 \approx 7$
- Hence, the computational efficiency ≈ 0.5/7 ≈ 10%
  - So, because of the bandwidth requirements, we have to buy 10 x more computing than a pure HPC system would require.
  - Unless the vendors improve the memory bandwidth…

Computing Requirements

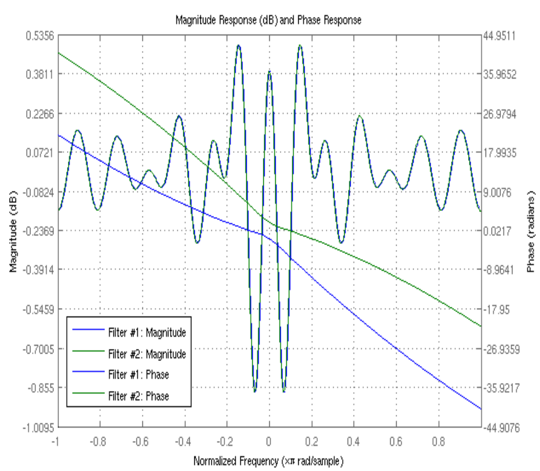
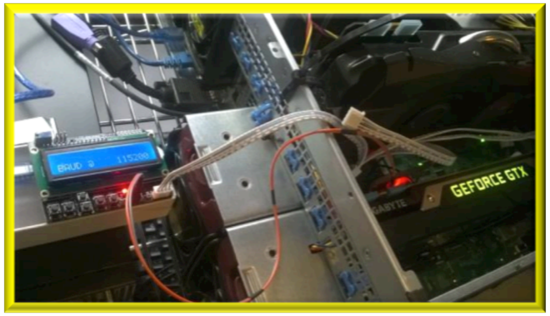
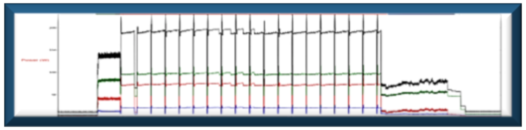
- ~100 PetaFLOPS/sec total sustained
- ~200 PetaByte/s aggregate BW to fast working memory
- ~50 PetaByte fast working storage
- ~1 TeraByte/s sustained write to storage
- ~10 TeraByte/s sustained read from storage
  - ~ 10000 FLOPS/byte read from storage
- Current power cap proposed is ~5MW per site.

Data Management Challenges

- All Top500 HPC systems have been designed for High Performance Computing (by definition).
- IDC has proposed a new term HPDA – High Performance Data Analytics to reflect systems like SKA
- Must ensure the data is available when and where it is needed.
- CPU's must not be idling waiting for data to arrive
  - Data must be in fast cache when it is needed.
- Need a framework that supports this.
  - Looking at a variety of possible prototypes

Addressing Power

- Need to achieve a FLOPS/Watt an order of magnitude better than current greenest computer.
- Need a three pronged approach:

Algorithms	**Hardware**	**Testing**
	 	 
Pursue innovative approaches to cut processing times	Look at accelerators, hosts, networks and storage.	Using real algorithms and fully instrumented systems

Software

- Budget of > €100M on manpower for software development across the whole telescope.
  - Not a task for academic programmers
  - Need professional practices for development, testing, integration and deployment.
  - Need to unify the processes across the world-wide team of developers.
  - Need world-leading expertise in a number of areas.
- Delivered system will not be static
  - SDP hardware and software will be updated periodically.
  - Key input for development will be the scientific and software community through the regional centres.

Conclusions

- SKA is a huge computational challenge
  - CSP ~ 50 Pflop, 5 MW
  - SDP ~ 100 Pflop, 5 MW
  - Tianhe-2 ~30 Pflop, 40 MW
- Traditional HPC is not a good match because the problem is bandwidth dominated.
  - SKA is seen as a key programme in global IT development
  - Showcases a major development area of High Performance Data Analysis (HPDA).
- Power is also a major driver.
- Software complexity is also beyond what has been achieved in astronomy previously.