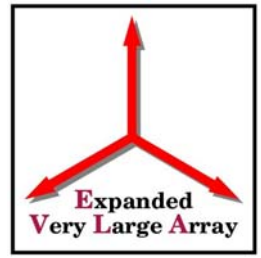
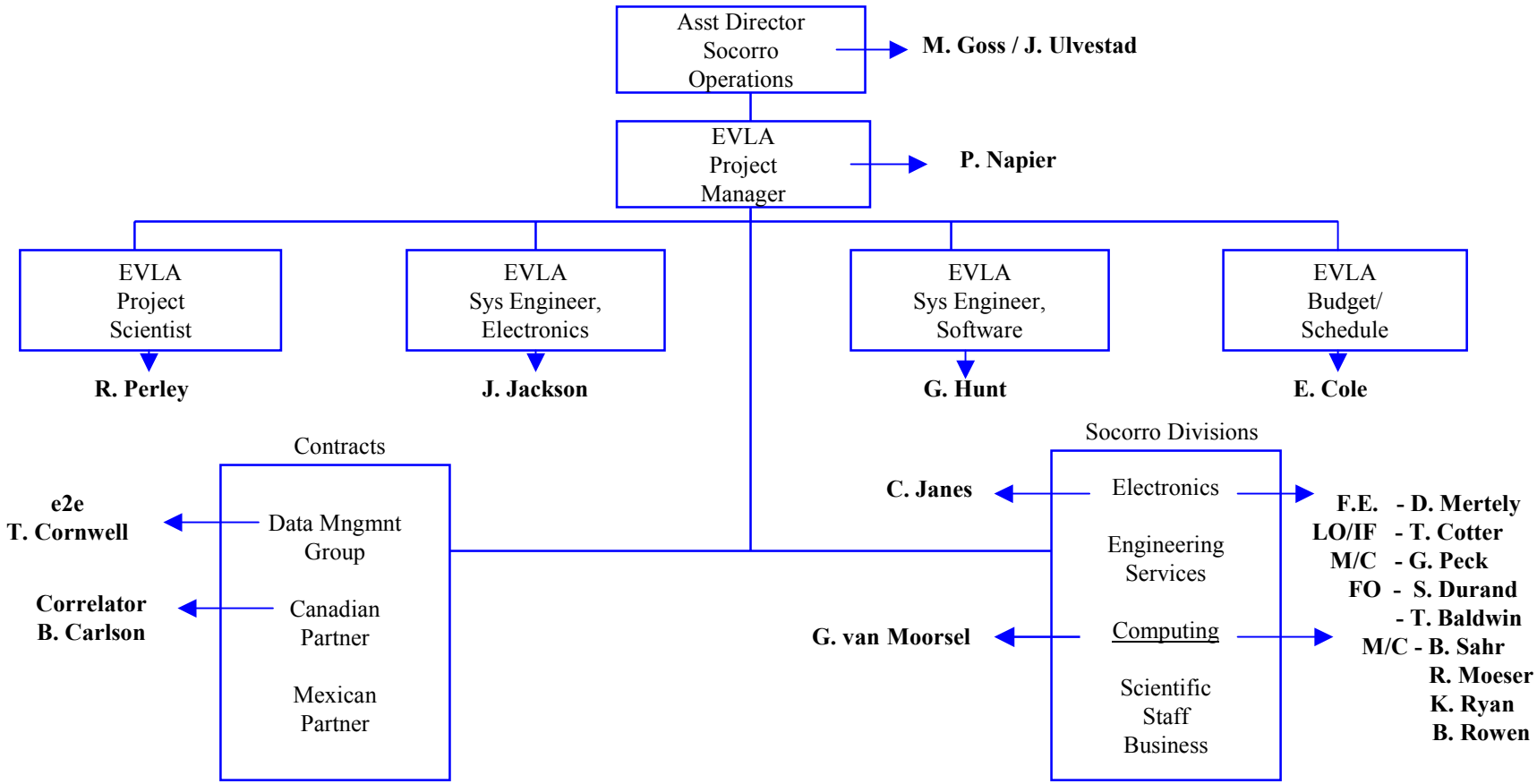
EVLA: Data Management

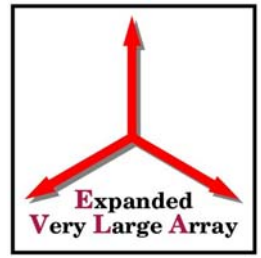


- EVLA sub-contracts data management to NRAO Data Management group
  - End-to-end processing needs being addressed by e2e project
  - Data reduction needs being addressed by AIPS++ project
    - Large data volumes, parallel processing
    - New processing needs *e.g.* wide-field, high dynamic range
    - Sub-band combination

EVLA management chart

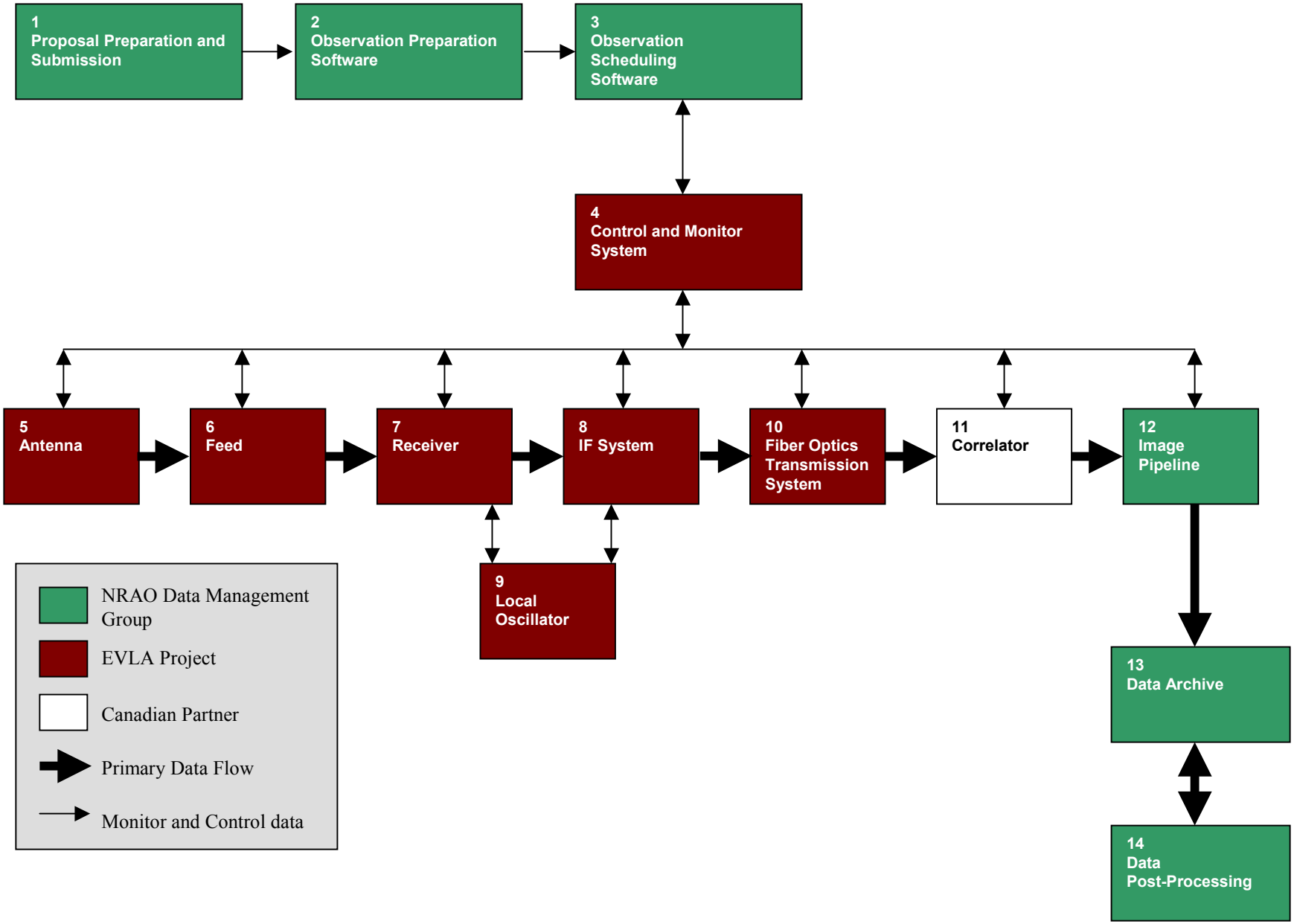


End-to-End Processing



- Is software development for end-to-end management proceeding satisfactorily?

EVLA data flow

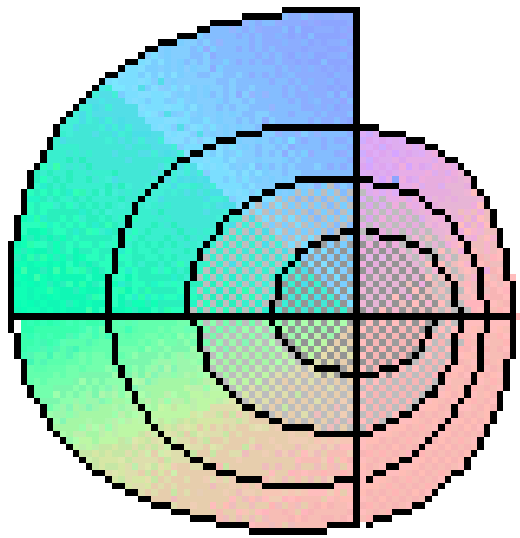


End-to-End project (e2e)

- End-to-End processing for all NRAO telescopes
  - Improve accessibility and usability of NRAO telescopes (VLA/VLBA, GBT, EVLA)
  - Build on and consolidate existing resources as much as possible *e.g.* AIPS++
- Development costs shared across NRAO
  - DM: project manager, project architect
  - Basic research: project scientist
  - Active construction projects: EVLA and ALMA
  - Sites (VLA/VLBA, GBT) and projects (AIPS++)
- Funding
  - Use internal contracts with EVLA, ALMA, GBT, VLA/VLBA
  - New collaborations: NVO, mini-COBRA
  - Have ~ 65 FTE-years
- Progress
  - Officially started July 1
  - Project book (http://www.nrao.edu/e2e)
  - Start slowly: entering phase 1 development: interim VLA archive and pipeline

Development

- Current staff
  - Tim Cornwell, Boyd Waters, John Benson
  - Job hiring in progress for C++/Java software engineer
  - 2 Pipeline developers soon (funded by ALMA)
  - Expect 6 – 8 developers by middle of 2002
- Use spiral development model
  - Five year development plan
  - Develop in 9 month cycles
  - Get requirements, plan, design, implement, test
  - Review requirements, plan, design, implement, test…..
  - Add new staff incrementally

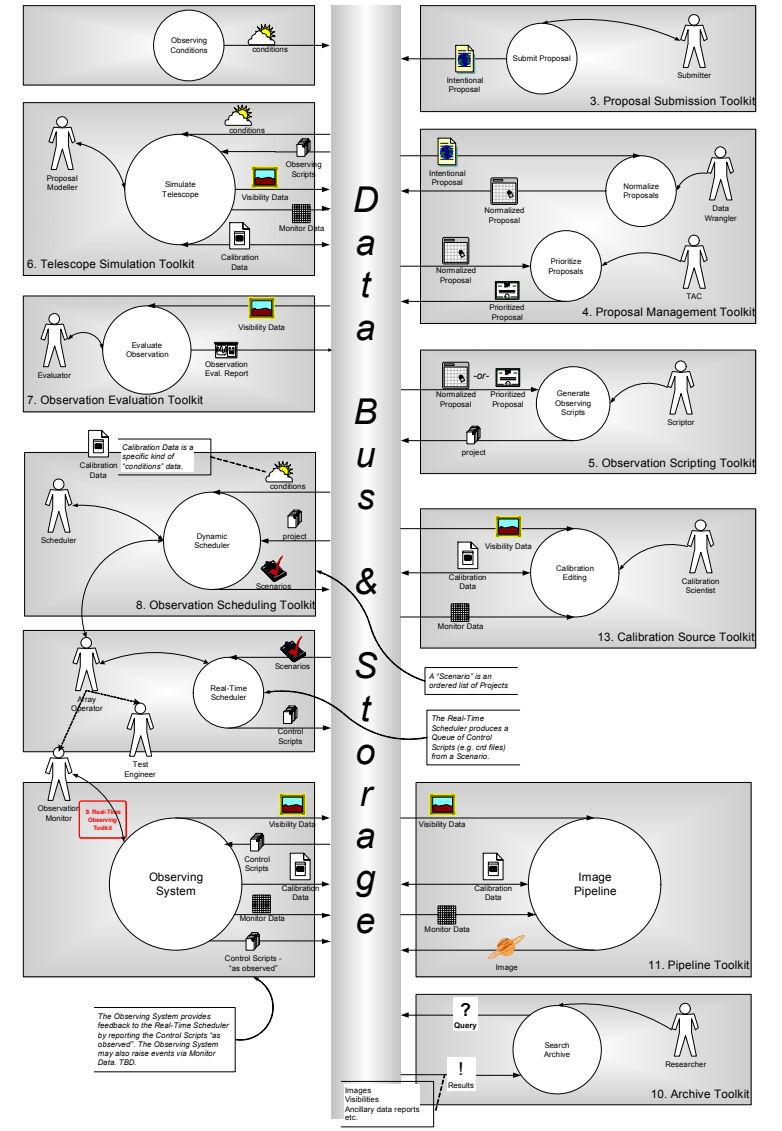


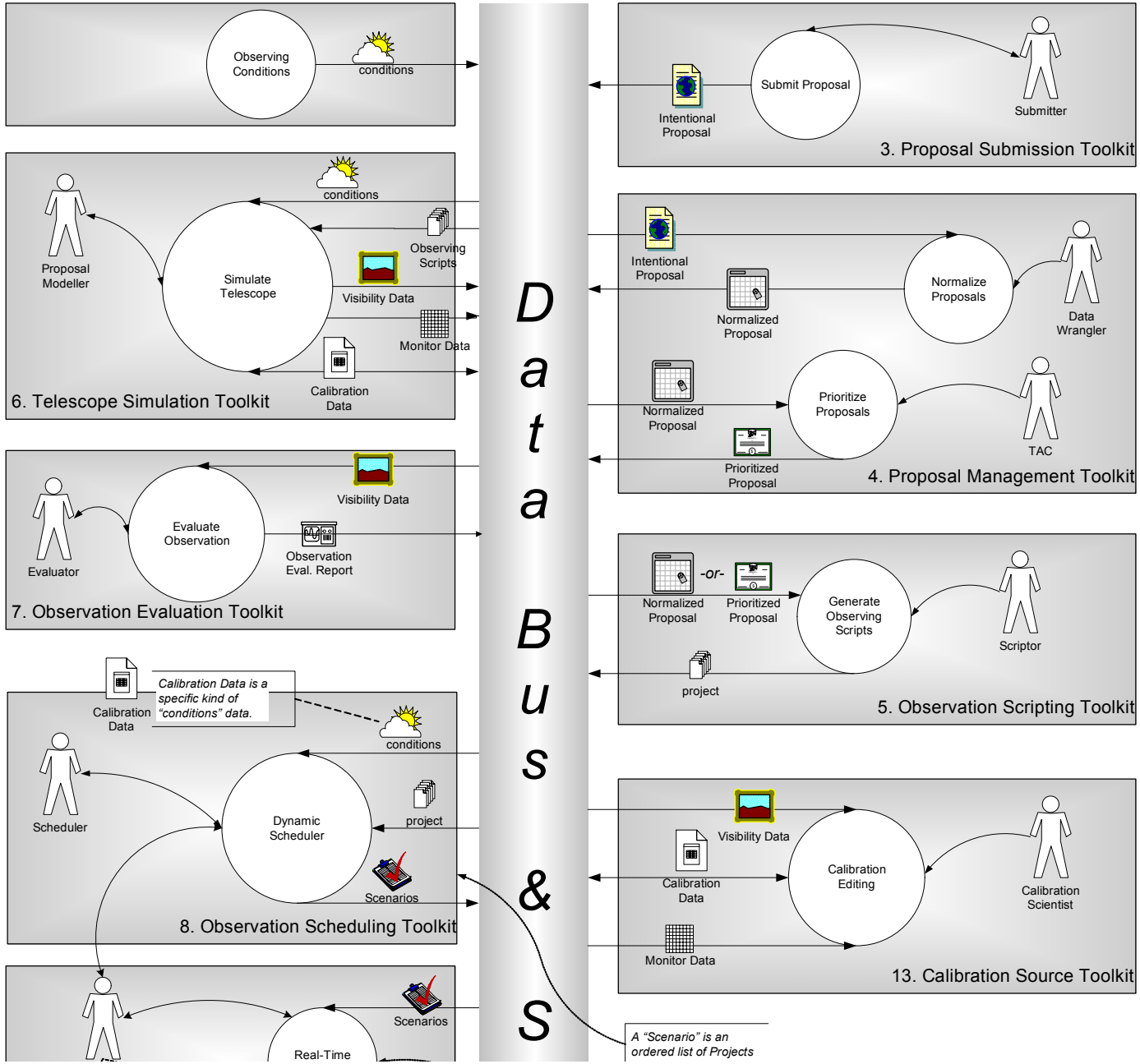
- First iteration: work on core of e2e
  - Interim VLA archive: get all VLA export tapes on line, investigate various archiving issues
  - Interim VLA pipeline: process some data from archive
  - Start initial development of scripting for observing and pipeline setup
  - Calibration source unification for VLA and VLBA

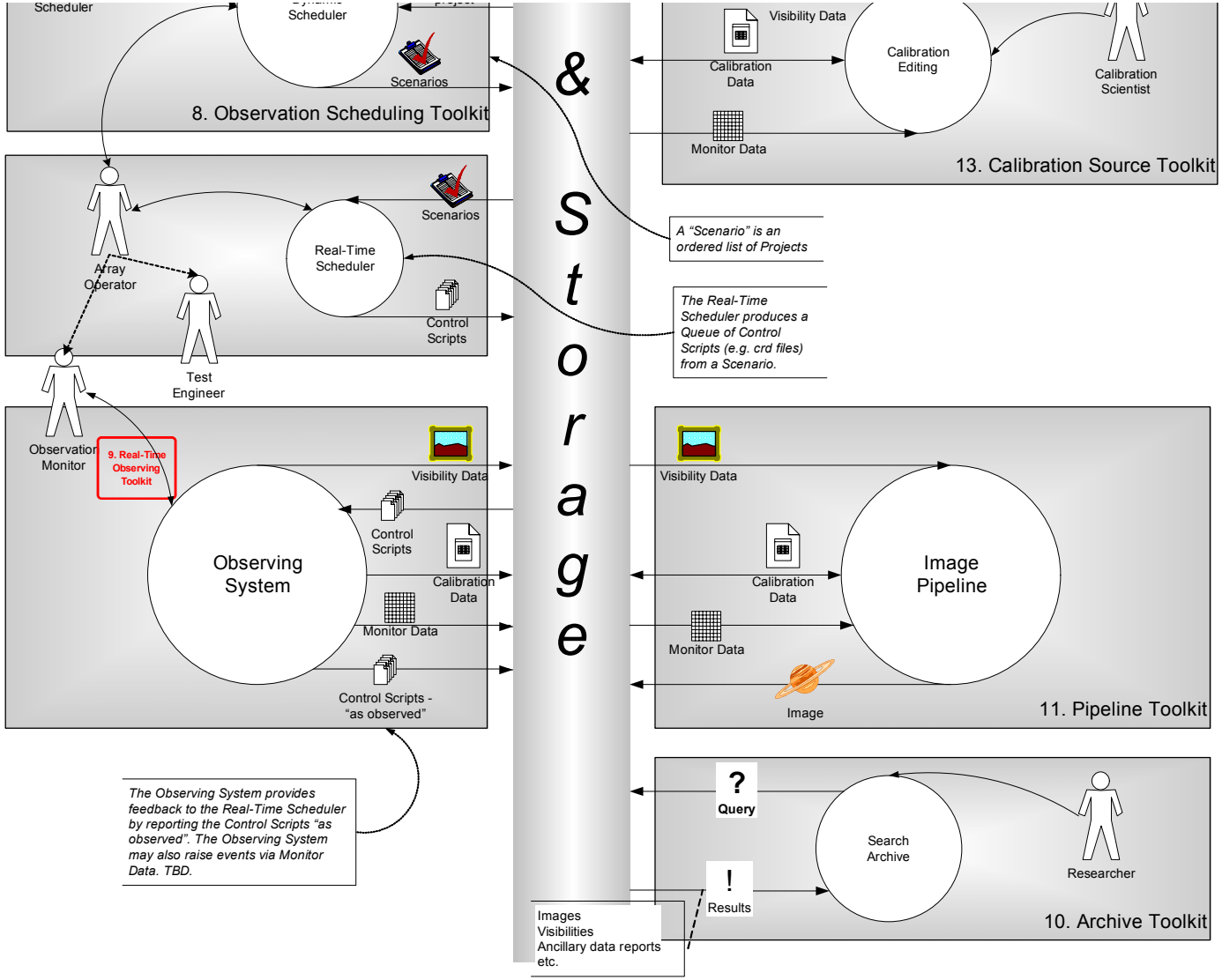
Development

- Extensive discussion of scientific requirements with Scientific Working Group
  - Captured in e2e project book
- Description of workflow from proposal to observing script
  - Converted to high level architecture and data flow
- Proceeding on basis of current requirements
- Revisit after ~ 9 months development of prototypes

e2e Architectural Diagrams




Observing Conditions
conditions
Data Bus & S
Intentional Proposal
Submit Proposal
Submitter
3. Proposal Submission Toolkit
conditions
Proposal Modeller
Simulate Telescope
Observing Scripts
Visibility Data
Monitor Data
Calibration Data
6. Telescope Simulation Toolkit
Intentional Proposal
Normalized Proposal
Normalize Proposals
Data Wrangler
Normalized Proposal
Prioritize Proposals
TAC
Prioritized Proposal
4. Proposal Management Toolkit
Visibility Data
Evaluator
Evaluate Observation
Observation Eval. Report
7. Observation Evaluation Toolkit
Normalized Proposal
-or-
Prioritized Proposal
Generate Observing Scripts
Scriptor
project
5. Observation Scripting Toolkit
Calibration Data
Calibration Data is a specific kind of "conditions" data.
conditions
Scheduler
Dynamic Scheduler
project
Scenarios
8. Observation Scheduling Toolkit
Visibility Data
Calibration Data
Calibration Editing
Calibration Scientist
Monitor Data
13. Calibration Source Toolkit
Scenarios
Real-Time
A "Scenario" is an ordered list of Projects



End-to-End: More Detail

Overall e2e architecture

Package	How?	Priority	Status
Operational Model	*Document*	*High*	*First version*
Proposal Submission Toolkit	*Web form or Java-based tool*	*Medium*	*Deferred*
Proposal Management Toolkit	*Java-based tools plus database*	*Medium*	*Deferred*
Telescope Simulation Toolkit	*AIPS++ tools*	*High*	*Deferred*
Observation Evaluation Toolkit	*AIPS++ tools*	*Medium*	*Deferred*
Observation Scripting Toolkit	*GBT Observe, GUI editor*	*High*	*Investigation*
Real Time Observing Toolkit	*Java, AIPS++ tools*	*Low*	*Deferred*
Observation Scheduling Toolkit	*OMS + local adaptations*	*Low*	*Deferred*
Archive Toolkit	*AIPS++ plus rdbms?*	*High*	*Prototyping*
Pipeline Toolkit	*AIPS++ tools*	*High*	*Prototyping*
Pipeline heuristics	*Glish scripts*	*High*	*Prototyping*
Calibration source toolkit	*OMS*	*High*	*In development*

Operational model

- Describes/prescribes operation of NRAO telescopes
  - Currently based on VLA/VLBA operational model
  - Will extend and make consistent with GBT
  - Yet to be agreed with telescope directors
- Covers
  - Proposal submission and management
  - Observing scripts
  - Scheduling of observations
  - Calibration and imaging
  - Interactive observing
  - Pipeline processing
  - Archive use
  - Quality assessment
  - Final products

Interfaces to EVLA M&C

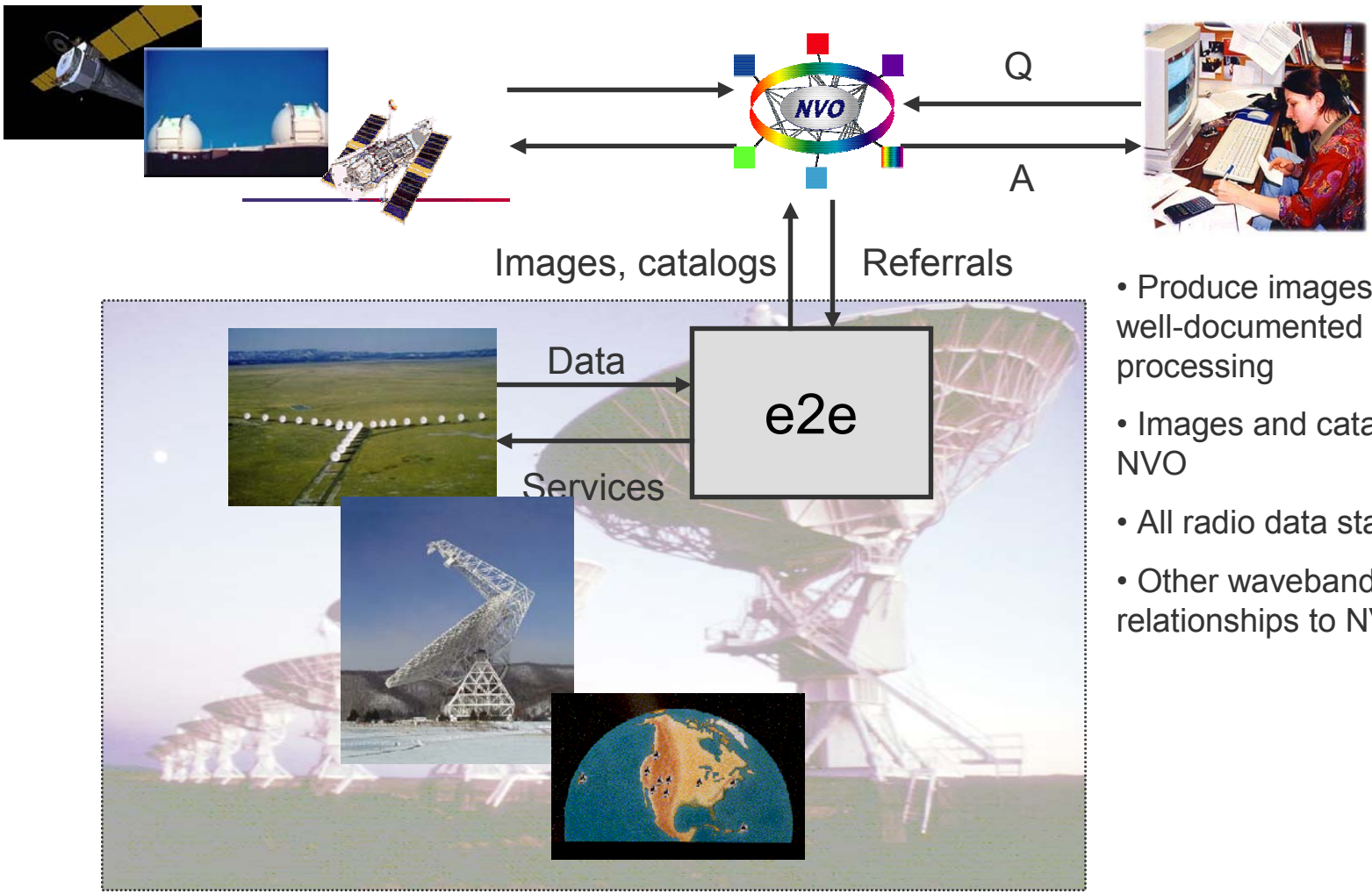
- Observing scripts:
  - Observing blocks ~ 20min duration
- Observed data:
  - Data in ~ AIPS++ MeasurementSets, one per observing block
  - Sent to archive by M&C
  - Evaluation by pipeline
- Calibration information:
  - *e.g.* antenna gains, baselines, *etc.*

Resources

- ALMA numbers estimated by ALMA computing management
  - Seem to be in line with other ground based projects
- e2e numbers based upon straw man designs, reuse
- e2e scope will be adjusted to fit resources (~ 65 FTE-years)
- Neither constitute a detailed bottom-up derivation of resources from requirements

Effort (FTE-years)	*ALMA*	*e2e*
Proposal Handling Software	14	10
Scheduling Software	8	15
Pipeline	12	10
Data Archive	12	20
Post Processing Software	11	10
Total	57	65

From NRAO to the National Virtual Observatory

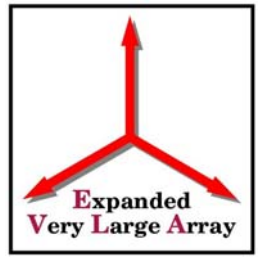


- Produce images and catalogs from well-documented pipeline processing
- Images and catalogs “sent” to NVO
- All radio data stays within NRAO
- Other wavebands have similar relationships to NVO

Data processing

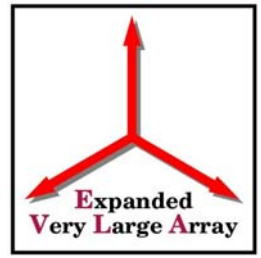
- Scale of processing: can it be handled by 2009-era hardware?

The numbers



- Peak data rate ~ 25 MB/s
- Data for Peak 8-hr observation ~ 700GB
- Floating point operations per float ~ 100 - 10000
- Peak compute rate ~ 5Tflop
- Average/Peak computing load ~ 0.1
- Average compute rate ~ 0.5Tflop
- Turnaround for 8-hr peak observation ~ 40 minutes
- Average/Peak data volume ~ 0.1
- Data for Average 8-hr observation ~ 70GB
- Data for Average 1-yr ~ 80TB

Detailed analysis



- Analyze processing in terms of FFT and Gridding costs
- Find scaling laws for various types of processing
- Express in terms of 450MHz Pentium III with Ultra-SCSI disk
- Use Moore's Law to scale to *e.g.* 2009
  - Performance/cost doubles every 18 months
- Many more details in EVLA Memo 24

Detailed analysis

tion	# pol	FOV arcsec	Cellsize arcsec	Pointings	Facets	Pixels	BW MHz	Freq res MHz	Vis chan	Image chan	IF's	T obs hr	T int sec	vis/int
y beam (2D)	4	7200	0.3	1	256	24000	500	1.00	500	1	1	12	3	702000
y beam (3D)	4	7200	0.3	1	1	24000	500	1.00	500	16	1	12	3	702000
GRA West	2	200	0.2	64	1	1000	70	0.5468	128	128	8	8	10	718848
galaxy	2	600	0.5	1	1	1200	7	0.006	1166	1024	1	24	10	818532

tion tion	Data rate Mb/s	Total data GB	Image Mpixel	Visibilities Mvis	Minor cycles	single d	multiple d	mosaic d	Time d	# processors	rate TB/year
y beam (2D)	1.87	80.87	576	10108.80	10	28.50	35972.08	40.88	35972.08	71944.16	59.04
y beam (3D)	1.87	80.87	9216	10108.80	10	130.48	194.08	232.88	130.48	260.96	59.04
GRA West	0.58	16.56	128	2070.28	100	19.97	296.85	34.20	34.20	102.59	18.14
galaxy	0.65	56.58	1679.04	7072.12	10	38.30	122.53	56.96	38.30	38.30	20.65

Scale of processing

- Assume Moore's Law holds to 2009
  - Moore himself believes this……..
- Cost of computing for EVLA
  - ~ 10 – 20 processor parallel machine
    - ~ $100K - $200K (2009)
  - Archive ~ 50TB per year
    - ~ $50K - $100K (2009)
- Comparable to computing cost for ALMA
- Software costs
  - AIPS++ *as-is* can do much of the processing
  - Development needed for high-end, pipelined processing
  - Some scientific/algorithmic work *e.g.* achieving full sensitivity, high dynamic range

Desktops vs. servers

- Moore's Law gives ~ 64 fold increase for a desktop
  - *I.e.* $nK where n ~ 1-3
- Many projects do-able on (2009-era) desktop
  - *e.g.* 1000 km/s velocity range of HI for galaxy
  - *e.g.* Mosaic of SGRA West in all H recombination lines between 28 and 41 GHz
- Larger projects may require parallel machine or many days on a desktop
  - *e.g.* Full sensitivity continuum image of full resolution 20cm field
  - NRAO would provide access over the net

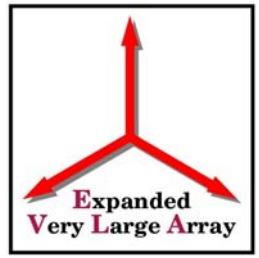
Data processing

- Is software development for data processing proceeding satisfactorily?

General AIPS++ performance

- Performance standards for AIPS++:
  - Must be comparable to other disk-based packages
  - If not, filed and handled as a high-severity defect
- Analysis of existing performance defects:
  - No inherent design-related problems found so far
  - Cases of poor performance have been invariably due to drift as part of regular code evolution
- Current approach to performance issues:
  - Have existing correctness tests which are run regularly
  - Building separate performance benchmark suite
  - Will run routinely to inter-compare AIPS++ and other packages, and catch performance drift early
  - Performance benchmarks will cover a wide range of problem sizes and types
  - Have a separate high-performance computing group within AIPS++

AIPS++ high-performance computing group

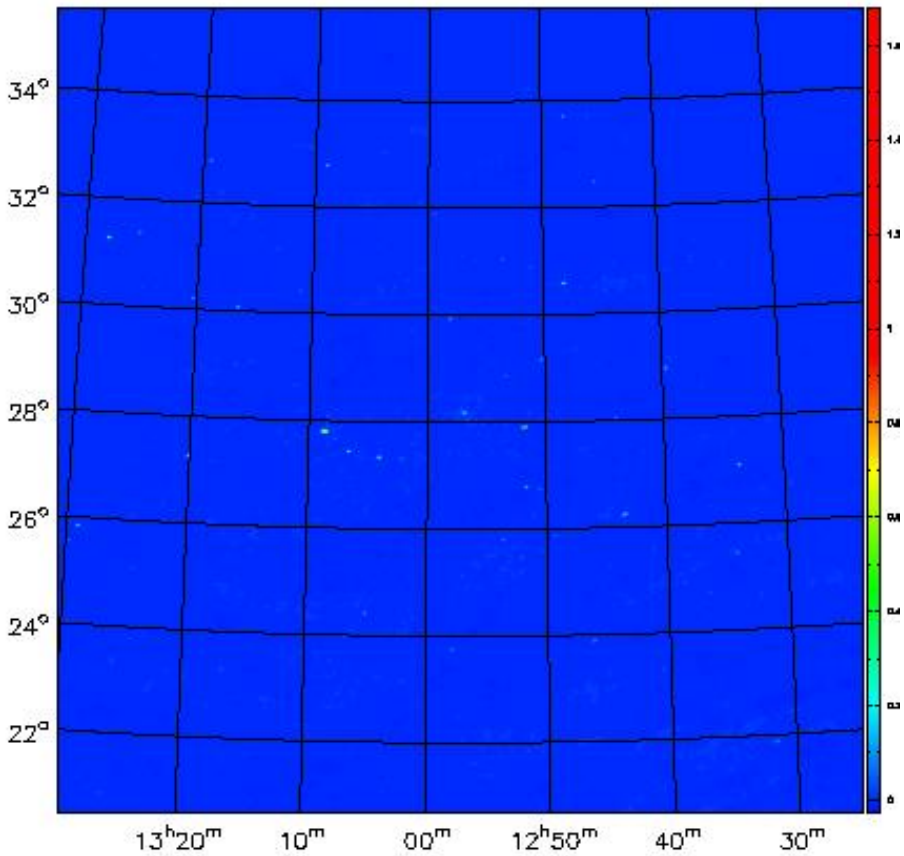


- Joint initiative with the National Center for Supercomputing Applications (NCSA) in Urbana-Champaign, as part of the broader NCSA Alliance program
- Separately funded by an NSF grant
- Objectives:
  - Address computationally challenging problems in radio astronomy which require supercomputer resources
  - Provide an AIPS++ infrastructure to integrate support for HPC applications
  - Provide portable solutions on common supercomputer architectures and Linux clusters
  - Build expertise in HPC issues such as parallel I/O, profiling and algorithm optimization

Example: parallelized wide-field VLA imaging

VLA observations of the Coma cluster (test data courtesy Perley *et al.*)

225 imaging facets, 32 processors, speed-up factor ~20 to a net 10 hours elapsed time



AIPS++ pipeline development

- Pipelines in AIPS++:
  - A key requirement across the consortium and affiliates
  - Prototypes (ATCA, BIMA) or full systems (ACSIS, Parkes multi-beam) underway
  - Design effort within AIPS++ and with other projects (e.g. ALMA)
- VLA prototype pipeline:
  - Under development as part of the first e2e prototype
  - Based on the 2 TB VLA disk archive to be deployed soon
  - Have purchased a pipeline server (4-processor Linux IBM x370 system) for the prototype pipeline system
  - Early version will be confined to very restricted VLA observing modes (likely continuum)
  - Prototype will test prototype pipeline design, implementation and performance issues on a short time-scale (Spring 2002)
  - Vital feedback for more complete pipeline design and development work for the VLA/EVLA

Post processing

- Mostly well-understood and in place
  - AIPS++ package: can reduce VLA data end-to-end
- EVLA-specific areas requiring more development
  - Very high dynamic range
  - Achieving full continuum sensitivity at 1.4 GHz and below
    - Asymmetric primary beams
  - RFI mitigation
    - ATNF post-correlation scheme
    - Masking, passive and active
  - Very large data volumes