

CASA and DMS System Design

Scope :

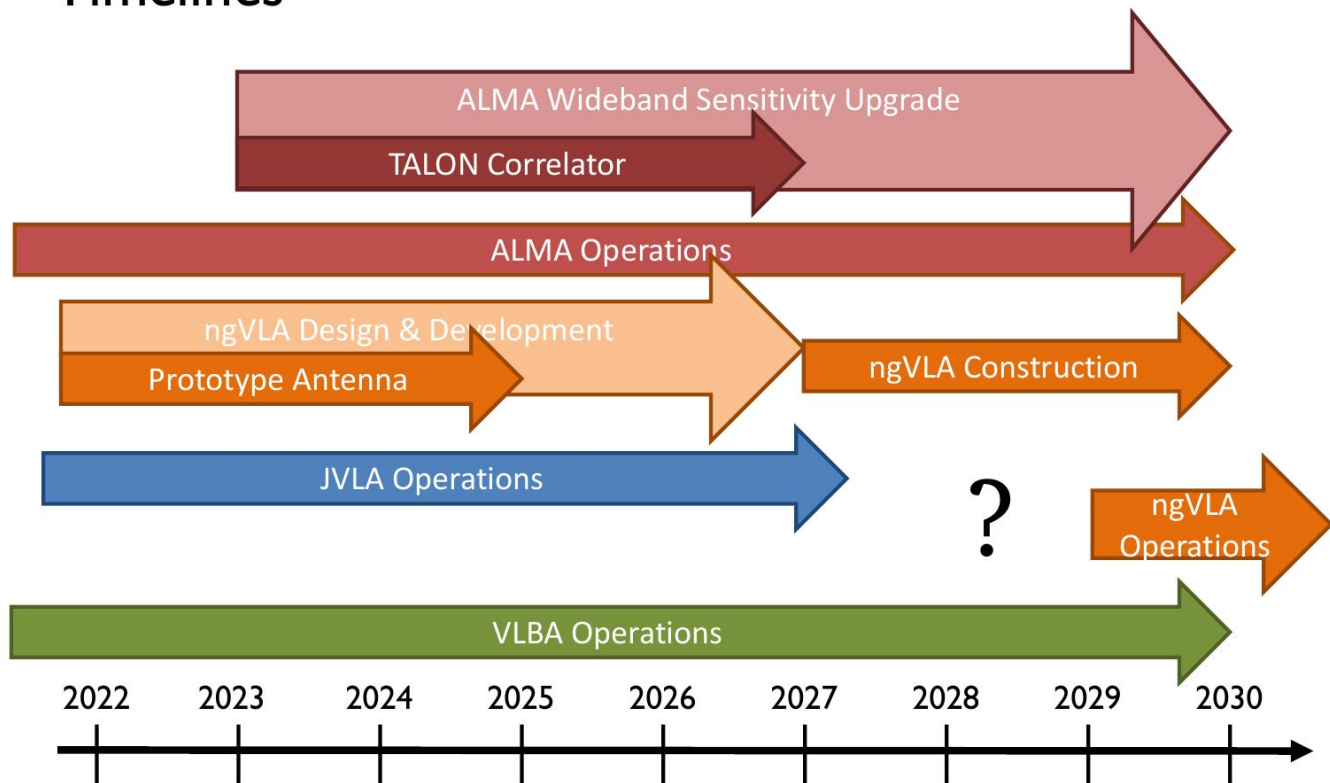
- Future of the CNGI project : Interaction with DMS-Systems for design of end-to-end processing workflows
- CASA's evolution towards supporting ngVLA and ALMA-WSU : New infrastructure and software
- Supporting current operations and stakeholders : Interoperability with current data and software

Summer 2022 : Initial discussions to define topics/timelines for analysis, before a design can be finalized.

Note : These slides capture our current thoughts, for the purpose of communication / discussion and feedback.

The plans will evolve along with our understanding, and are not final in any way.

Timelines



ngData Processing Architecture Roadmap

	DMS		ALMA		ngVLA	
FY22 Q3			June 21,22	Internal WSU Concept Review		
FY22 Q4			Sept ?	External WSU Concept Review		
FY23 Q1	~Nov 1		NRAO Internal Review			
FY23 Q2			~Feb/March	WSU Concept Delta Review	~Feb/March	Updated Cost Estimation
FY23 Q3					~March/April	Software CDR Programmatic System CDR
FY23 Q4						
FY24 Q1					~March	System PDR

NRAO Internal Review Objectives

Demonstrate the scale and implications of the data processing challenge.

- Size of Computing Memo
- Technical Use Cases
- Data Stores Technical Requirements
- Our concept for addressing the issues
- Data Processing Conceptual Architecture
- A prototyping plan to support the FY23 reviews
- Derisk key implementation choices



[DMS Systems Design Plans](#)

- ⇒ CASA participation is needed
- Definition
 - Execution

Ongoing DMS-Systems work

Topics relevant to CASA (prioritized)	Timeline	CASA staff @0.2FTE	Status (Fall 2022)	DMS Projects
1. Prototyping of Processing Framework (dask/cngi/xarray for casa7 : yes or no ?)	Fall 2022 (to decide..) 2023? (to design..)	DMS-Sys + ~3-4 casa	Ongoing : Evaluation of CNGI framework against end-to-end system requirements. <ul style="list-style-type: none"> Several questions have been identified. A small group is being assembled to start prototyping. 	ngData Processing Software Architecture
2. Data Quantum / Model / Format / API (to decide API for ongoing ngcasa dev – interface between subsystems)	Fall 2022 (ms-support : how?) 2023+ (new formats)	DMS-Sys ~3-4 casa	On hold : Convertors/readers for the MSv2 for interoperability with casa6. <ul style="list-style-type: none"> Waiting on a choice of framework 	Data Model Development
3. Algorithm & Pipeline Architecture (to decide algorithm feature set and granularity of the programmer API : CASA + PIPE)	Fall 2022	DMS-Sys + ~2 casa	Ongoing : Brainstorming on Algorithm Architecture <ul style="list-style-type: none"> Pipeline workflows with calibration and imaging as iterative solvers Identify the right level of granularity of functional building blocks 	ngData Processing Algorithms Concept
4. CASA7 API (to start ngcasa dev.)	Fall 2022 (initial version)	CASA	Not started : Derive a CASA7 API, based on the above building blocks.	
5. Science Use Cases, Metrics, Specs (to assess feature completeness and write functional/stakeholder/performance tests)	Fall 2022 (for compute sizing) 2023+ (for features)	DMS-Sys ~1-2 casa & stakeholders	Ongoing : Write/gather ngVLA and ALMA-WSU use-cases / specs <ul style="list-style-type: none"> Focus on bracketing use-cases that will drive the technical design. 	Requirement Refinement
6. User Interaction Requirements/Specs? (to design user-interfaces : gui, api, nb)	2023+	DMS-Sys + ~1 casa	Not started : UX concept does not exist yet. <ul style="list-style-type: none"> DMS is in conversation with stakeholders to start this thought. 	User Interaction Concepts

Data Model/Format/API and Parallelization Frameworks (DMS: Data Model Development)

Data Quantum :

Shape/size of each data block
E.g. One SPW, one timestep

10^n way parallelization
→ $> 10^n$ way partitioning.
(depends on use-case)

Data Model :

MS.v.3 (or 4?)
ASDM++

Include metadata required by
archive (atomicity, consistency, isolation, durability),
pipeline bookkeeping, processing.

Data Format (on disk) :

Requirements :

- What is the shared data format for Online System, Data Archive and Post-Processing ?
- Compatibility with existing software ?

Example Options :

MS (casacore tables, perhaps with Adios) : Portability
ASDM-like binary : Online systems already use it
Zarr/NetCDF : Native integration with XArray/Dask

Programming APIs/Frameworks

To/From data

- CASA tools and C++ access
- XArray

Parallelization

- MPI, OpenMP, Cuda....
- Dask (w/ Cuda)

Pipeline Operations

System Prototyping to decide framework choices ? [i.e. Continue the CNGI project, but with DMS ?]

=> Make some initial (informed) choices for the above topics. Test & Refine.

E.g. Can we decide if we can (or not!) add new CASA methods from 2023, using MS(v2) <-> XArray with Dask parallelization, where the XArray internally holds a list of XDSs (one per ddid) ?

Algorithm, CASA, Pipeline Architecture

(DMS: Data Processing Software Processing Arch)

CASA library API/Specification

Goal : A consistent level of granularity across all modules.
(One level ? Two levels ? More ? CASA/Pipe?)

Example....

```
flag_manual(data_selection_params),
flag_auto(algo_choices)
```

```
cal_solve(sol_type)
cal_apply(caltables)
```

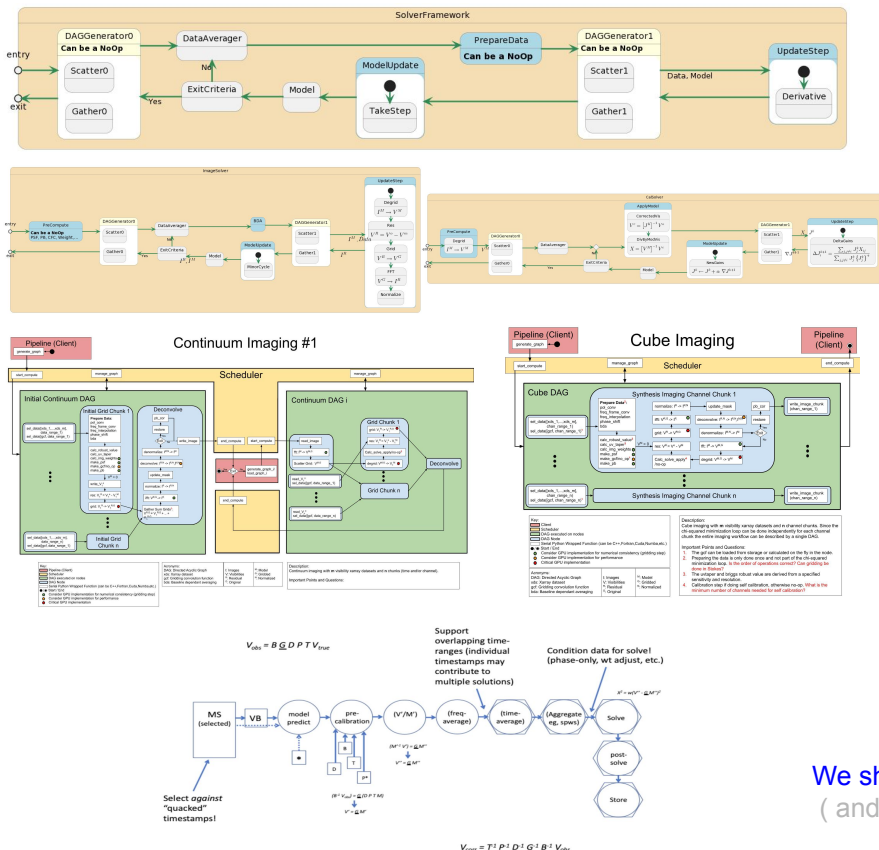
```
make_psf(weighting_options)
make_pb(pbmodel_options)
vis_to_image(gridding_algo_choices, imshape),
model_to_vis(degridding_options)
make_mask(mask_options)
deconvolve(sky_model, algo_choices, iteration_control)
restore(beam_options)
pb_correct(image, pb)
```

```
vis_average(freq_bin, time_bin)
vis_phaseshift(target_dir)
vis_doppler_correct(target_frame)
```

```
pipe_imageprecheck(), pipe_calc_some_heuristic(),....
Astropy functions for some things ?
```

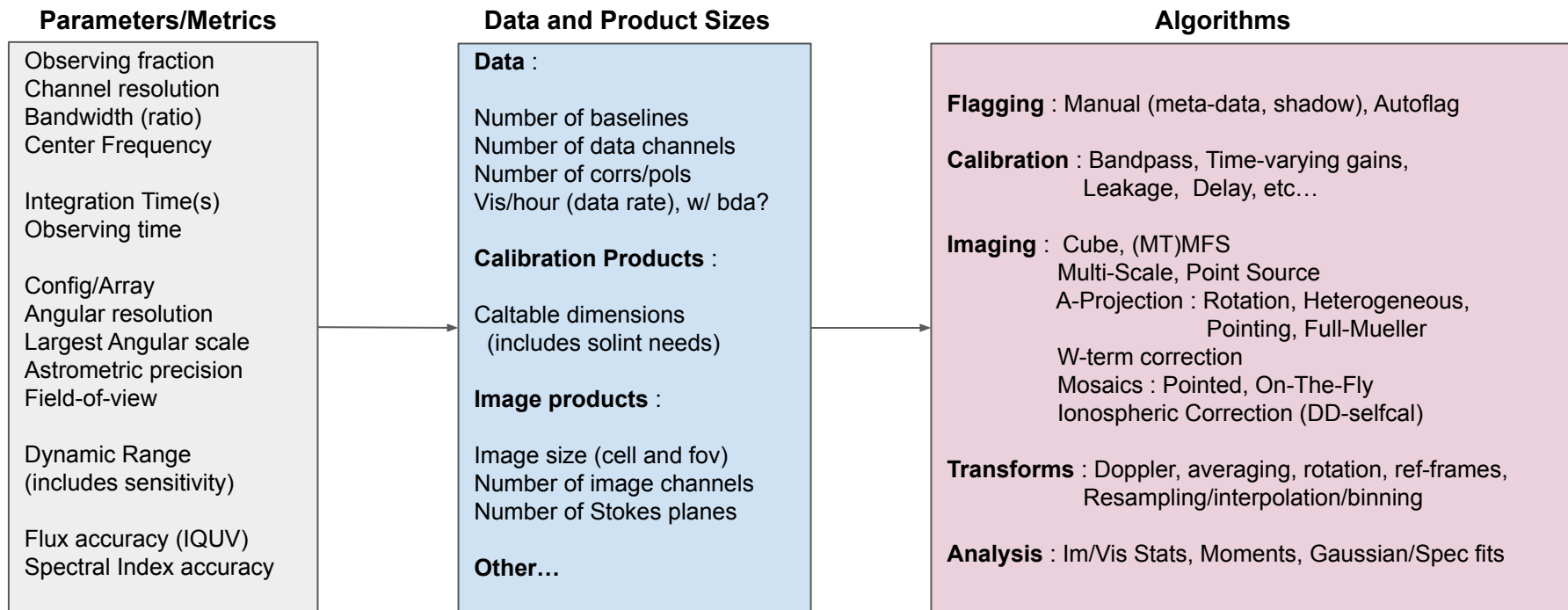
We should (approximately) define this API, before we start writing ngCASA methods
(and not directly start copying the casa6 task-list....)

Also, think about User-facing (telescope/pipe specific documentation) vs CASA lib docs.



Examples of diagrams from various people....

Science Use Cases (DMS: Requirement Refinement)



Purpose : Derive bounds on numerical accuracy of functions + Information for stakeholder & performance tests to assess operational readiness.
Work is ongoing to write this for ngVLA + ALMA-WSU. Stakeholders are writing the use-cases. We are helping define the format of the info....
Include system verification (numerical, using fiducial error/accuracy values with propagation) as part of the 'system prototyping' effort ?

References : [Tables 1,2,3 from the ngVLA Size of Computing Memo](#) (including reference and envelope Science UseCases) + [DSA-2000 Science-To-Data example](#) + [VLASS Regs/Specs](#) + [ngVLA System Requirements](#)

Operational Use Cases & User Interaction (DMS: User Interaction Concepts)

Use Case	What to visualize? What to interact with ?	User Interface	Responsible ?
Pipeline Standard Reduction Non-interactive	Output images/cubes Image analysis : Spectra/Moments	Workspaces (dashboard, GUIs)	DMS SSA : front end CASA/Pipe : backend
Interactive Pipeline Re-Processing 'User Driven'	Output & Intermediate images/cubes Plot calibration solutions and visibilities Summary views/plots from pipelines ? Interactive clean mask drawing ? ? Interactive flagging or views for autoflag tuning ?	Workspaces (dashboard, GUIs, Jupyter notebooks)	DMS SSA : front end CASA/Pipe : backend
Development & Commissioning Manual Reduction	All images, interactive masks, autoflag results, caltables... Visibility scatter/raster plots/views Interactive clean mask drawing, interactive flagging, etc... Meta-data based views (e.g. ant location plots)	Terminal + Scripts (scripts, notebooks, headless GUIs?)	CASA (simplified toolset using python/scripts) 3rd party packages

References : [ngVLA SysArch docs](#) for reduction vs analysis + [casaqui sys design docs](#) + [Workspaces](#)

How much of this should CASA need to do ?

DMS 5-year planning → For CASA, this is the CASA6 to CASA7 transition.

These are only initial ideas. They will change based on DMS-wide scheduling, staffing, stakeholder needs, and practical feasibility...

Topic	FY23	FY24	FY25	FY26	FY27	FY28
DMS Systems ngData processing architecture design	<ul style="list-style-type: none"> - Infrastructure prototyping and scaling demo. - Data Model/Format - Algorithm/Software architecture (casa/pipe + functional API) - Reqs/specs for bracketing use-cases 	<ul style="list-style-type: none"> - Data Quantum - Pass sim data through skeleton ? - Reqs/specs fill in use-cases and add metrics - User-interaction concept/reqs 	<ul style="list-style-type: none"> - Data model version 1 specification 			
CASA7 ngData processing implementation Features and algorithms	<ul style="list-style-type: none"> - Starter cngi projects for dev training : feather, msuvbinflag, imstat, holog,panel/locfit : + ngVLA prototype Antenna commissioning - GPU dev training 	<ul style="list-style-type: none"> - Convertors and readers for (new?) data format - Continue starter cngi projects for dev training - GPU dev with cngi imager prototype 	<ul style="list-style-type: none"> - Prototype apps for cube imaging, calibration, simulation, flagging, VLBI, singledish - Target use-cases from current telescopes +ALMA un-mitigated cube imaging +VLASS full stokes GPU 	<ul style="list-style-type: none"> - Development against API and use-cases/metrics for ALMA-WSU and ngVLA - Support for pipe dev/ops + ALMA un-mitigated cube imaging + VLASS full stokes GPU - Engage with ARDG to bring algorithms into new framework 	<ul style="list-style-type: none"> - Development against API, use-cases, metrics for ALMA-WSU and ngVLA 	<ul style="list-style-type: none"> - Development against API, use-cases, metrics for ALMA-WSU and ngVLA
CASA6 Operations support for current use cases	<ul style="list-style-type: none"> - Support for pipeline ops +ALMA Band1, self-cal... - Continue VLBI dev - Interactive Clean 	<ul style="list-style-type: none"> - Support for pipeline ops + re-calibration, group imaging? - Continue VLBI dev - Deprecate viewer 	<ul style="list-style-type: none"> - Maintenance and bugfixes only 	<ul style="list-style-type: none"> - Maintenance and bugfixes only 	<ul style="list-style-type: none"> - Maintenance and bugfixes only 	<ul style="list-style-type: none"> - Maintenance and bugfixes only

Relevance to VLBI-CASA development

Input from this group will be required to refine our concept of how VLBI development should evolve.

- Continue development for VLBI support : [Long-term stakeholder requirements](#)
 - New features/algorithms
 - AIPS -> CASA transition of required features.
- Design efforts in FY23
 - Include VLBI use-cases in the CASA/DMS use-case documents
 - VLBI-specific methods to be included as part of the overall CASA7 functional API specification
 - Participation from this group : Use-case definition and review of design plans.
- CASA6 -> CASA7 transition, starting in mid FY23 and/or FY24
 - CASA7 : Start adding new features directly here. Start transfer of existing methods from CASA6.
 - CASA6 : Bug fixes and minor feature additions only
 - CASA6 and CASA7 will have to be interoperable until CASA7 has all required features.

For example,

```
from casatasks import gaincal, applycal
from casangi import holog
gaincal(vis='xxx.ms',...)
applycal(vis='xxx.ms',...)
holog(vis='xxx.ms',...)
```

Some points for discussion :

- For new VLBI features in FY23/24, how feasible is it to develop standalone methods ?
 - Goal : To avoid having to do it twice.
 - Examples : For comparison, refer to plans for 'feather', 'msuvbinflag', 'holog', 'panel', 'imstat', etc....
- What is most efficient to do in the casa6 code base ?
- Fraction of VLBI use-cases and features that will be covered by ngVLA ?
- ??

Notes

....