

DRAFT: Workspaces System Implementation Planning

- Introduction
 - Planning Process
 - Planning Guidelines
- Phases
 - Infrastructure Phase 0: Refactor Existing Software Repositories/Build Systems
 - Affected Components:
 - Subtasks
 - Infrastructure Phase 1: Initial Skeleton
 - Affected Components:
 - Subtasks
 - Infrastructure Phase 2: Simple Download Capability and Metrics
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 1: Initial User Interface Prototype
 - Infrastructure Phase 3: Authentication, Authorization and Allocation (A3)
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 2: Simple Download Capability
 - Infrastructure Phase 4: Standard Calibration Capability, Transformations & Restores
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 3: Simple Download with A3
 - Infrastructure Phase 5: Preliminary Large Project Support
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 4: Transformations & Restores
 - Infrastructure Phase 6: Search Filter & Results Authorization
 - Affected Components:
 - Subtasks
 - Infrastructure Phase 7: Ingestion
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 5: Search Filters & Results
 - Infrastructure Phase 7B: Reingestion
 - Infrastructure Phase 8: ALMA Downloads
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 6: QA Assignment Interface
 - Infrastructure Phase 9: ALMA Restores
 - Affected Components:
 - Subtasks
 - Infrastructure Phase 10: AUDI (ALMA User Driven Imaging)
 - Affected Components:
 - Subtasks
 - Infrastructure Phase 11: VLASS
 - Affected Components:
 - Subtasks
 - Browser Interface Phase 8: User Interface for AUDI
- Decision Tracking
- Unresolved Questions (to be resolved before we complete this document)
- Risk Register

Introduction

This document and accompanying spreadsheet describe our implementation plan for the workspaces system.

Planning Process

- We will define two series of milestones to build the system up incrementally; the first series are Infrastructure Phases and the second series are Browser Interface Phases. Browser Interface Phases are staggered behind the Infrastructure Phases they are meant to expose and exercise.
- Phases will have well-understood deliverables that are small enough to be meaningful for tracking the project and no smaller.
- We will decompose these phases into pieces of work using the tee shirt sizing system: S (1 day), M (3 days), L (5 days) for effort estimation.
- As part of this decomposition we will note dependencies between pieces of work.
- Work that is too large to fit into a week will be further decomposed into pieces that do fit into a week or less.
- We will address work on tests and documentation as part of our development process itself: these will not be tracked as standalone work.
- Once we have the complete breakdown we will build a schedule: the schedule itself **is not** a deliverable for this Implementation Planning, but the developer assignments **are**.

Planning Guidelines

- We will use a "walking skeleton" approach and initially produce naive implementations of all the system's components sufficient to get the system working end-to-end in a limited fashion.
- We will build on this skeleton and add features and functions as we go.
- We will build testing infrastructure into the project from the beginning, with API validation and regression tests for each capability we implement.
- Each capability will be delivered with a command line interface that exercises it to facilitate testing.
- Each capability will be delivered with a regression test.
- Where we can, we will focus on replicating the functions of the existing implementation before addressing new features. Where we have identified risk we will mitigate it as early as possible.

Phases

Infrastructure Phase 0: Refactor Existing Software Repositories/Build Systems

We do some cleanup and reorganization of the existing archive software repository and structure before we begin development on the workspaces system in earnest, splitting the software in the existing repositories up into repositories for archive infrastructure, workspace infrastructure, common models, common libraries, command line interfaces and the browser interface.

Affected Components:

Subtasks

Task	Assignee	Size	Rationale
Pyat/Services Refactor	JS, RL, RF	L	Need to decide pieces
Delivery Task Refactor	JG	L	
Split 5 Repos	Team	L	Whole team has to meet; executing decision will be short
Kill RH	DL	M	
Kill OODT	DL	M	

Infrastructure Phase 1: Initial Skeleton

We will implement the skeleton of the workspaces system and its testing suite. All components of the workspaces system will be in place with naive implementations and enough functionality to execute a Testing Capability via command line interface; this Testing Capability will execute on the cluster and in OpenScienceGrid through HTCondor submission, to allow us to test capturing and reporting different execution results (exiting non-zero, exiting due to being killed by a signal, exiting successfully, and so on). This phase also includes preliminary implementation of the build and deployment system.

Affected Components:

1. Scheduling
2. Templating
3. Notification
4. Workflow
5. Capability
6. Build System
7. Deployment System

Subtasks

Task	Assignee	Size	Rationale
Productionize Prototype	DL	L	Nearly a walking skeleton already
Build system - not UI	JP	L	Research into Conda / building needed
Build system - UI	RG, RF	M	Conda questions
Deployment system	CH	L	
Process changes	DL, RL	S	Decisions to be made next week

Infrastructure Phase 2: Simple Download Capability and Metrics

We will add functionality to the skeleton to support downloads and delivery of data stored in the New Mexico (AOC) NGAS cluster; this stage won't include authentication or authorization, so it will be internal only. As part of the work we will capture the information needed for system metrics. This phase also won't include transformation of the data, just downloads.

Affected Components:

1. Capability

Browser Interface Phase 1: Initial User Interface Prototype

We will create a prototype of the user interface, starting with archive search and transporting products into the workspace system, based on interface definitions created in Phase 1. This prototype will have the basic outline of the major parts for simple use cases, like following the update progress of an item through a simple workflow.

Task	Assignee	Size	Rationale
------	----------	------	-----------

2. Workflow
3. Metrics

Subtasks

Task	Assignee	Size	Rationale
Data Fetcher	JG	L	
Delivery	JG	M	
Metrics	JP, RG	M	Questions about InfluxDB
Download Capability	CH	M	
Templating	JS	M	

Infrastructure Phase 3: Authentication, Authorization and Allocation (A³)

Same as phase 2, but the request fails if the data is proprietary (or a processing request) and the user either didn't provide credentials, provided bad ones or isn't authorized for the data. At this stage Allocation will be a stub; allocation checks will always return true, but the checks will be made.

Affected Components:

1. A3
2. Notification

Subtasks

Task	Assignee	Size	Rationale
Notification	JS	M	
Authentication	RF	L	Ensure we have ALMA stubs
Authorization for Proprietary Data	JP	M	Ensure we have ALMA stubs
Estimation Service	JP	M	
Scheduling Service	CH	S	
Scheduled Clean-up Tasks	CH	M	
Cancellation	DL	S	

Infrastructure Phase 4: Standard Calibration Capability, Transformations & Restores

We will add the VLA standard calibration capability to our system via command line interface, as well as restores of previously calibrated VLA data sets and the ability to deliver a basic MS. This stage brings the CASA Pipeline and PPR generation into the system.

Affected Components:

1. Product Service
 - a. Future Product
2. Workflow Service
 - a. Standard Workflows
 - i. Standard Calibration
3. Capability Service
 - a. Capability Info
 - i. Capability Matrix
 - b. Capability Queue
 - c. Capability Request
 - i. Request Version
4. Templating
5. Scheduling

Subtasks

Skeleton UI	RG	M	
Dummy launcher	RG, RF	S	
Skeleton UI Services	RF	M	
Research push technology	RF	M	Web sockets, ASGI, etc.

Browser Interface Phase 2: Simple Download Capability

We will provide the graphical interface components for starting and tracking of a download request by integrating with the workspace infrastructure.

Task	Assignee	Size	Rationale
Changes to search UI	RG, RF	M	
Request status page	RG	S	
Request UI service	RF	S	
Implement push technology	RF	M	

Browser Interface Phase 3: Simple Download with A³

We will add authentication check feedback in the browser interface for download requests. This should include helpful messaging about requests that cannot be processed and why. Controls for authentication and login will also be created.

Task	Assignee	Size	Rationale
Login button	RG	S	
Welcome UI service	RF	S	JWTs
Display proprietary status	RG	S	
Capability feedback messaging	RG	S	
Request history service	RF	S	
Request history UI	RG	S	

Task	Assignee	Size	Rationale
Standard Calibration, Restore Capabilities	CH	M	
CASA Wrapper	JG	L	
PPR Gen	JS	L	
Restore Prep	JS	S	
Request Version	JP	M	
Capability Matrix / CASA Version	DL	L	

Infrastructure Phase 5: Preliminary Large Project Support

We will provide an initial implementation of Large Project Support; executing the VLA Standard Calibration capability on a VLASS execution block will result in a calibration produced with a VLASS specific PPR template. This isn't intended to be production worthy, it is intended to exercise the functionality.

Affected Components:

1. Capability Service
 - a. Capability Info
 - i. Project Settings
2. A3
3. Notification Service
4. Templating
5. Scheduling

Subtasks

Task	Assignee	Size	Rationale
VLASS Calibration Capability	CH	M	
Stress Test Capability	CH	L	
Capability Info/Project Settings	DL	L	
A3 support	JP	L	What is this besides the next item?
Authorization - Restricted Capabilities	JP	S	

Infrastructure Phase 6: Search Filter & Results Authorization

We will implement the infrastructure to provide a search service the front end will use to allow users to query for requests with the following limitations:

1. Unauthenticated users won't be allowed to query or see results
2. Authenticated users will be able to query their own requests.
 - a. By date, capability, request status
3. Operations users will be able to query all requests and user operations type filters
 - a. As above, plus by requesting user or assigned user

Affected Components:

1. Capability Service
 - a. Capability Info
2. A3

Subtasks

Task	Assignee	Size	Rationale
Request Search	RF	L	

Browser Interface Phase 4: Transformations & Restores

We will add graphic controls to the browser interface to start and track VLA standard calibrations along with restoring previous calibrated VLA data sets and basic MS delivery.

Task	Assignee	Size	Rationale
Forms	RG	M	
Follow-on Requests	RG	S	
Cancel UI	RG	S	
Show Estimate	RG	S	

Infrastructure Phase 7: Ingestion

We will implement the ability for the workspaces system to ingest new products similarly to the way it does now, and we will implement the infrastructure for VLA Execution Block ingestion to trigger the VLA Standard Calibration capability.

Affected Components:

1. Ingestion Workflow
2. Reingestion Capability

Subtasks

Task	Assignee	Size	Rationale
Ingestion Workflow	JS	S	
Standard Calibration + QA	CH	S	
Ingestion Tools	JS	L	Could be moved to Phase 0
Ingestion Refactor	RL	L	Could be moved to Phase 0
Triggered Ingestion /Calibration	DL	L	May require additional architecture

Infrastructure Phase 7B: Reingestion

Task	Assignee	Size	Rationale
Formalizing reingestion & version requirements	SW	L	
Product System Reingestion Research	RL, JS	L	In need of analysis
Ingestion Refactor (reingestion)	RL	M	
Reingestion Tools	JS	M	
Reingestion Workflow	JS	S	
Reingestion Capability	CH	M	
VLBA import (formerly reingestion)			In need of analysis
ALMA import (formerly reingestion)			In need of analysis
ALMA calibration import (formerly reingestion)			
Use cases for each existing type of product			

Infrastructure Phase 8: ALMA Downloads

We will add ALMA to our A³ system and add the ability to retrieve ALMA products directly to our system, removing our reliance on ALMA's asdmExportLight utility.

Affected Components:

Subtasks

Task	Assignee	Size	Rationale
ALMA Data Fetcher	JG	L	
Locator Service	DL	M	
A3 Service - ALMA support	JP	L	

Browser Interface Phase 5: Search Filters & Results

We will add role and user level aware filter options to the workspace browser interface.

Task	Assignee	Size	Rationale
Forms	RG	M	
Service layer support	RF	M	

Browser Interface Phase 6: QA Assignment Interface

We will add additional tools and views to the workspace browser interface for user with a QA role, specific to facilitating the 'QA pass' and 'QA fail' status of products and requests.

Task	Assignee	Size	Rationale
Assign analyst UI	RG	S	
Assign analyst service	RF	M	
Pass/Fail buttons	RG	S	
Edit/resend request	RG	S	
Link to weblog	RG, RF	S	

ALMA Delivery	JG	L	
---------------	----	---	--

"Assigned to" filter	RF	M	
Add/update QA notes	RF	M	

Infrastructure Phase 9: ALMA Restores

We will replicate the functions of the existing AAT/PPI to allow to restore calibrated ALMA MOUSEs by command line.

Affected Components:

Subtasks

Task	Assignee	Size	Rationale
ALMA PPR gen for restores	JS	M	
ALMA Restore Capability	CH	M	
ALMA Data Fetcher extension	JG	M	
Product Locator	DL	M	

Infrastructure Phase 10: AUDI (ALMA User Driven Imaging)

We will implement the existing AUDI functionality as a capability with a command line interface.

Affected Components:

Subtasks

Task	Assignee	Size	Rationale
AUDI Capability	CH	M	
PPR gen for AUDI	JS	M	
Help Desk Service	RF	M	

Infrastructure Phase 11: VLASS

We will implement scaffolding to allow VlassMngr to function with the workspaces system as it now does with the AAT/PPI: we will 1) trigger events for ingestion complete events, and 2) allow VlassMngr to trigger ingestions by event.

Affected Components:

Subtasks

Task	Assignee	Size	Rationale
VLASS ingestion trigger	JS	S	
VLASS notification events	JS	S	

Browser Interface Phase 8: User Interface for AUDI

We will add the user interfaces for the AUDI capability.

Task	Assignee	Size	Rationale
Forms	RG	M	Risky size estimate
Help desk UI	RG	M	

Decision Tracking

- We expect that for this project, Browser Interface functionality is best implemented in a *phased* manner that tracks the implementation of the workspaces infrastructure, but Browser Interface presentation is best implemented in an *agile* manner. Therefore, we will focus on functionality and track that development as we do the infrastructure work, but develop the presentation as we go, and if the stakeholders are not satisfied by the end of the project we will continue to iterate on it.
- VLASS:
 - We will support VLASS by implementing Large Project functionality.
 - VLASS Workflows: we will convert the VLASS workflows into Capabilities and refactor the VlassMngr interface so it tracks VLASS specific concepts and creates workspace requests.
 - VLASS Ingestions: currently ingesting VLASS products is handled by the AAT/PPI, we will address them using Large Project Support and we will provide a way for VLASS to trigger ingestions using the workspaces system the same way it does now as a shim until we can refactor the VLASS Manager and Workflows.

- Implementation planning for Large Project support **is** within the scope of this document. Implementation planning for the changes to VLASS workflows and VlassMngr **is not**. Fully integrating VLASS into the workspaces system will be a subsequent project with its own planning and milestones.
- We are aware the archive system needs to handle versioning. We are detailing the use cases and in the early requirements capture stage of that project; it will not be part of this implementation plan.

Unresolved Questions (to be resolved before we complete this document)

- We need a gap analysis to make sure that at the end of the project we are feature complete.
- Where in the phases do we allow authenticated and authorized users to abort or re-launch their requests?
- Where does the estimation service come into play?
- Only staff can set delivery locations. How is this handled with A3?

Risk Register

Title	Description	Likelihood	Impact	Mitigation
Knowledge Transfer	Three developers are transitioning to new roles, shifting 50% of their time into architecture, project management, and testing; as a consequence, the remaining developers are taking on systems and services they are not familiar with.	High	Medium /Schedule slip	Factor in the time needed as much as we can
JWT /Ticket granting tickets	We will be using JWT to attach authentication information to capability requests as they flow through the system: we know this is possible but haven't done it before.	Medium	Medium /Schedule slip	Accept the loss of functionality: pass JWT around but don't check it for now.
Federated A & A	For a dozen years or more the NRAO has been considering implementing a federated authentication scheme: there is a risk such a decision will be made and the Workspaces System won't be compatible. We have done what we can by choosing technologies that interface with most of the popular systems that provide federation, but until a decision is made, we have a risk.	Low	Medium	Accept the risk.