

Data Management Architecture

Kian-Tat Lim, DM Software Architect



Large Synoptic Survey Telescope

C → D

DM's Mission

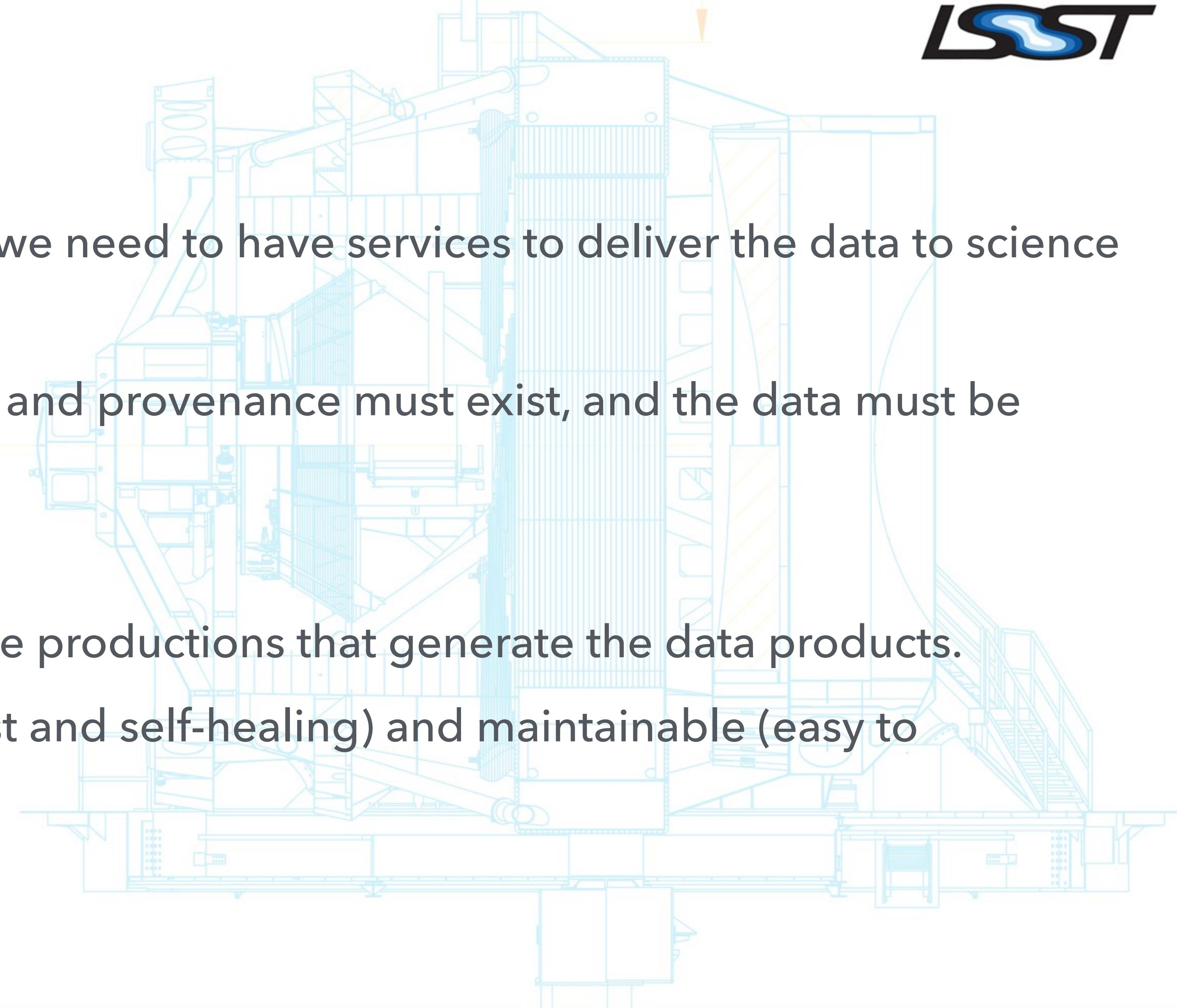


Stand up operable, maintainable, quality services to deliver high-quality LSST data products for science, all on time and within reasonable cost.



Services

- Producing data is not enough; we need to have services to deliver the data to science users.
- This means sufficient metadata and provenance must exist, and the data must be properly organized.
- Services also are used to run the productions that generate the data products.
- These must be operable (robust and self-healing) and maintainable (easy to reconfigure).

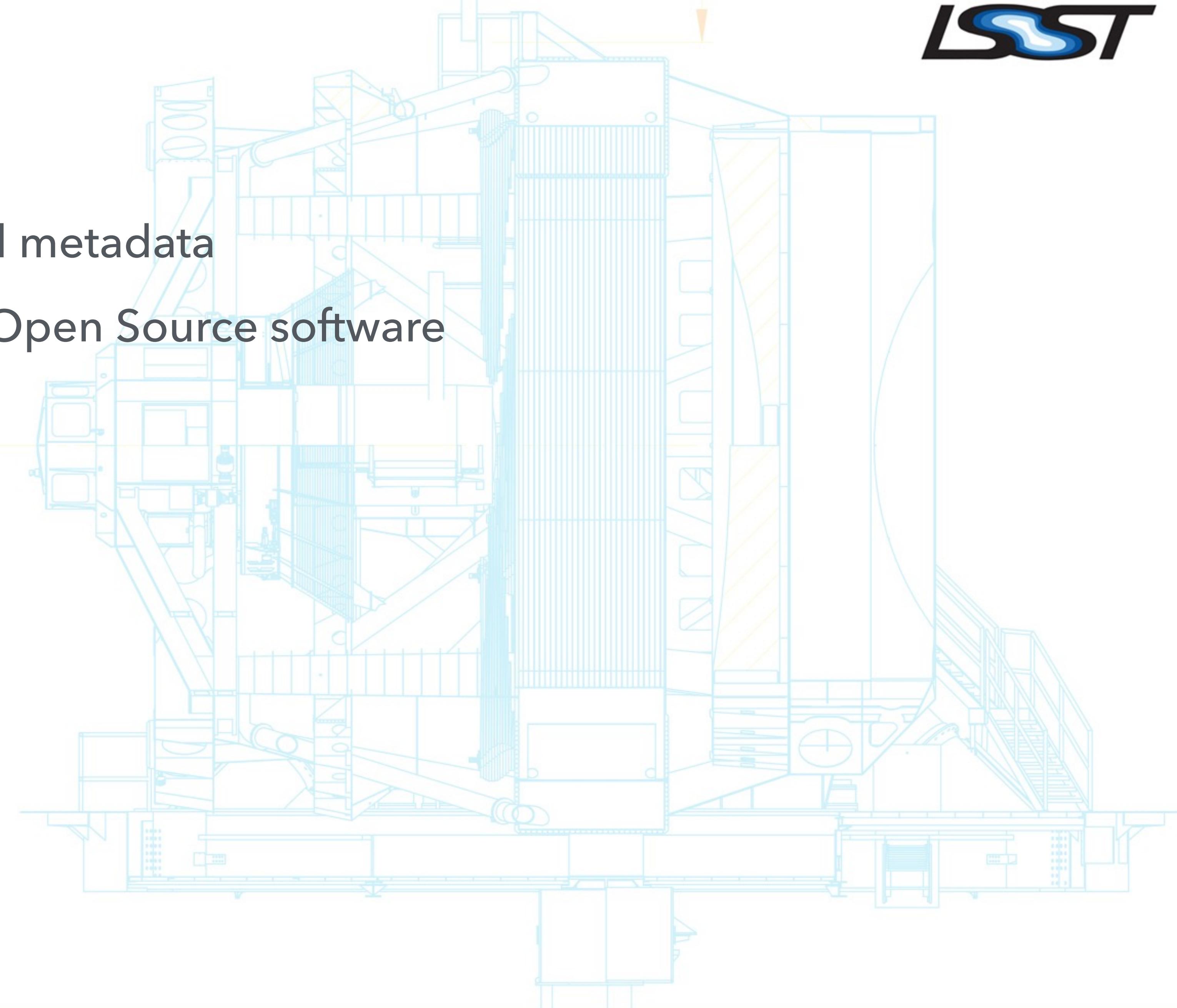


What We Do



LSE-61 (DM Requirements)

- Accept and archive images and metadata
- Generate data products using Open Source software
- Provide science user services



Accept and Archive Images and Metadata



LSE-68, LSE-69, LSE-72, LSE-130, LSE-140, LSE-400 (Interface Control)

- Images
 - Science
 - Calibration
 - Engineering
- Metadata
 - Auxiliary Instrumentation (LSST Atmospheric Transmission Imager and Slitless Spectrograph = LATISS, all-sky camera, weather, calibration instruments)
 - Commands, events, telemetry in the Engineering and Facilities Database

Generate Data Products



LSE-163 (Data Products Definition), LDM-151 (Algorithms Design)

Prompt:

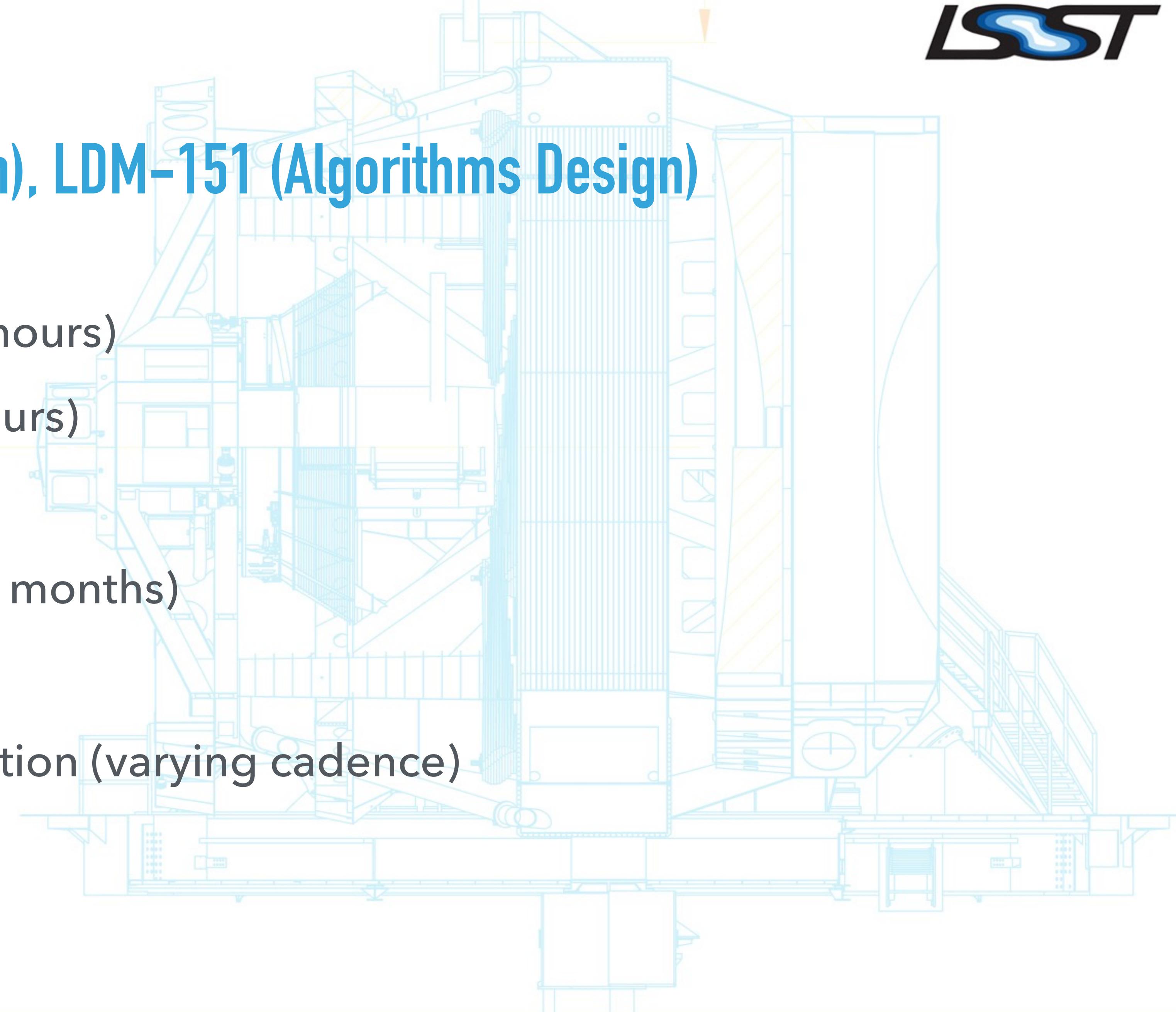
- Alert Production (60 sec, 6-24 hours)
- Solar System Processing (24 hours)

Data Release:

- Data Release Production (6–12 months)

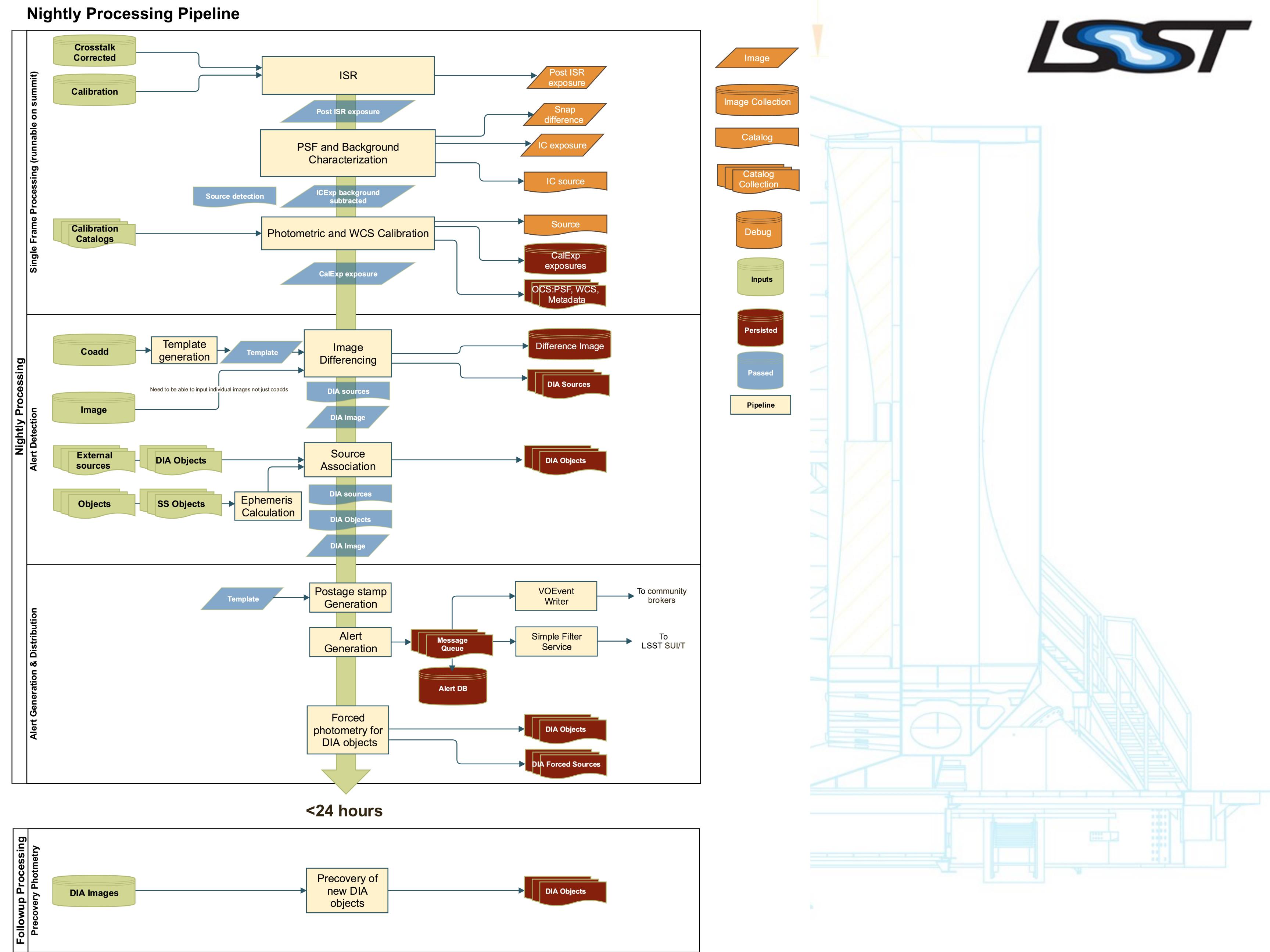
Calibration Products:

- Calibration Productions Production (varying cadence)



Alert Production

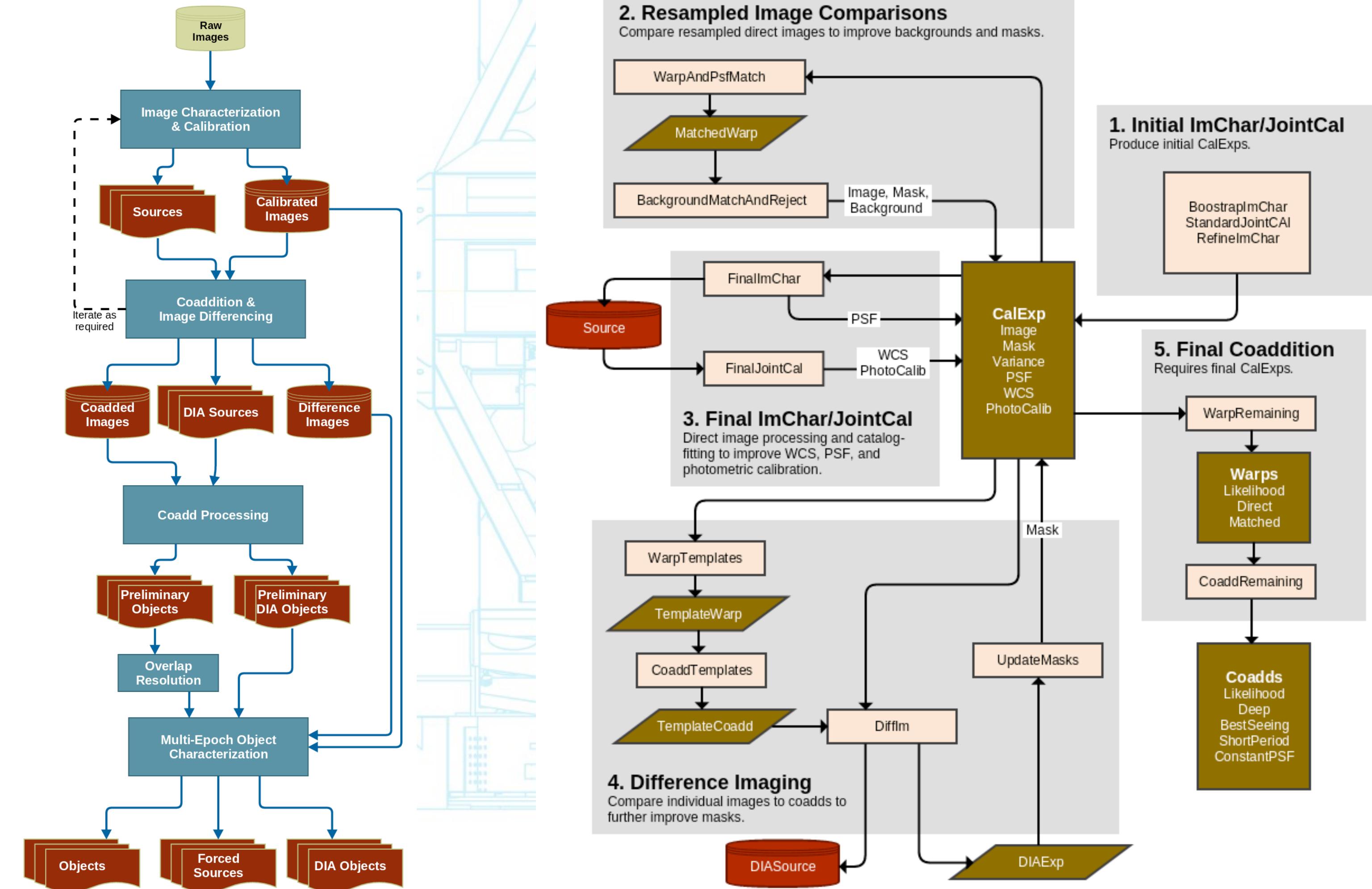
LDM-151



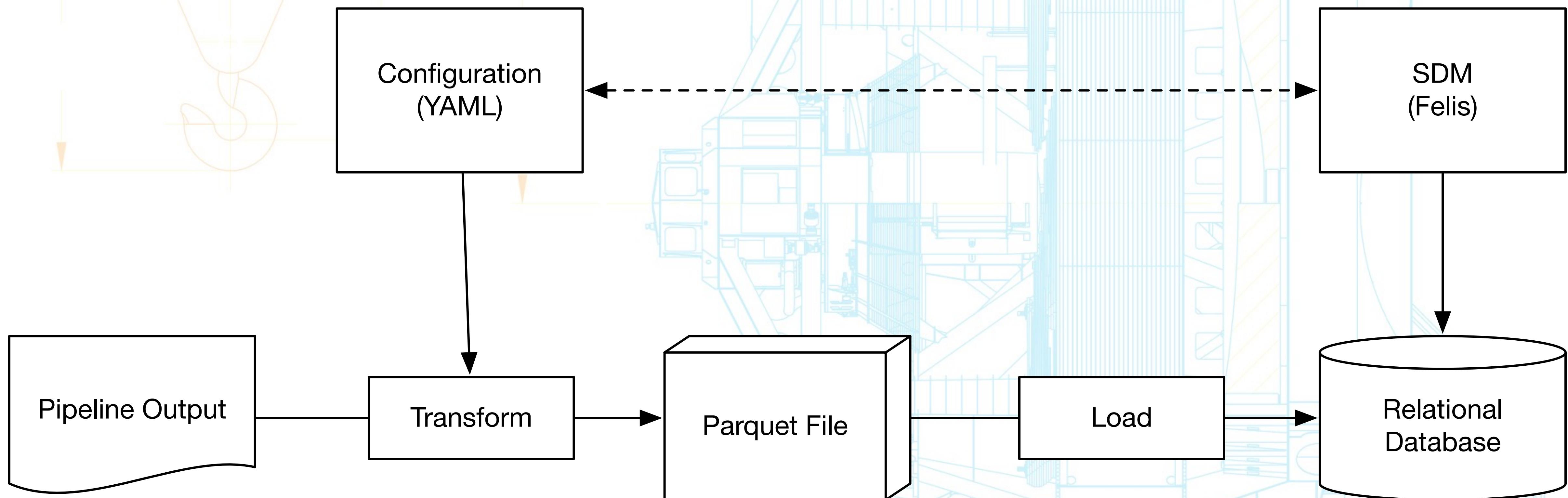
Data Release Production



LDM-151



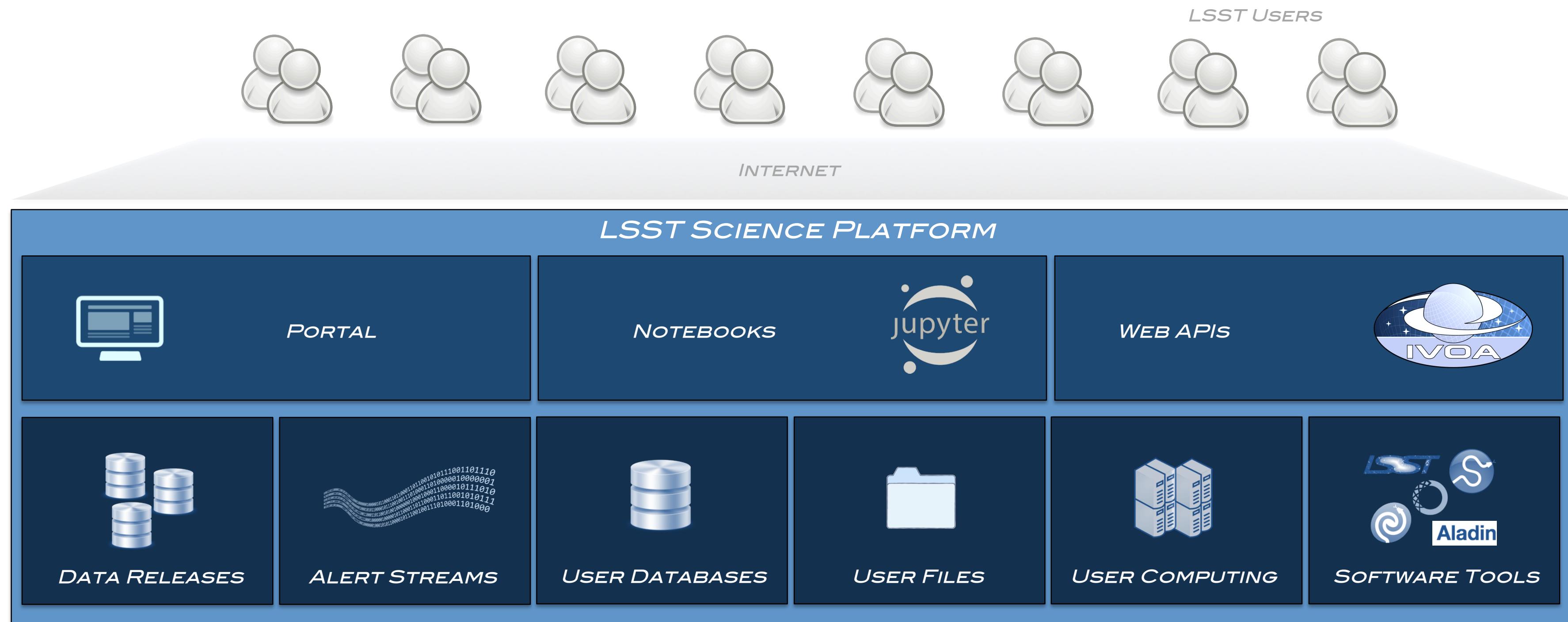
Science Data Model



Provide Science User Services



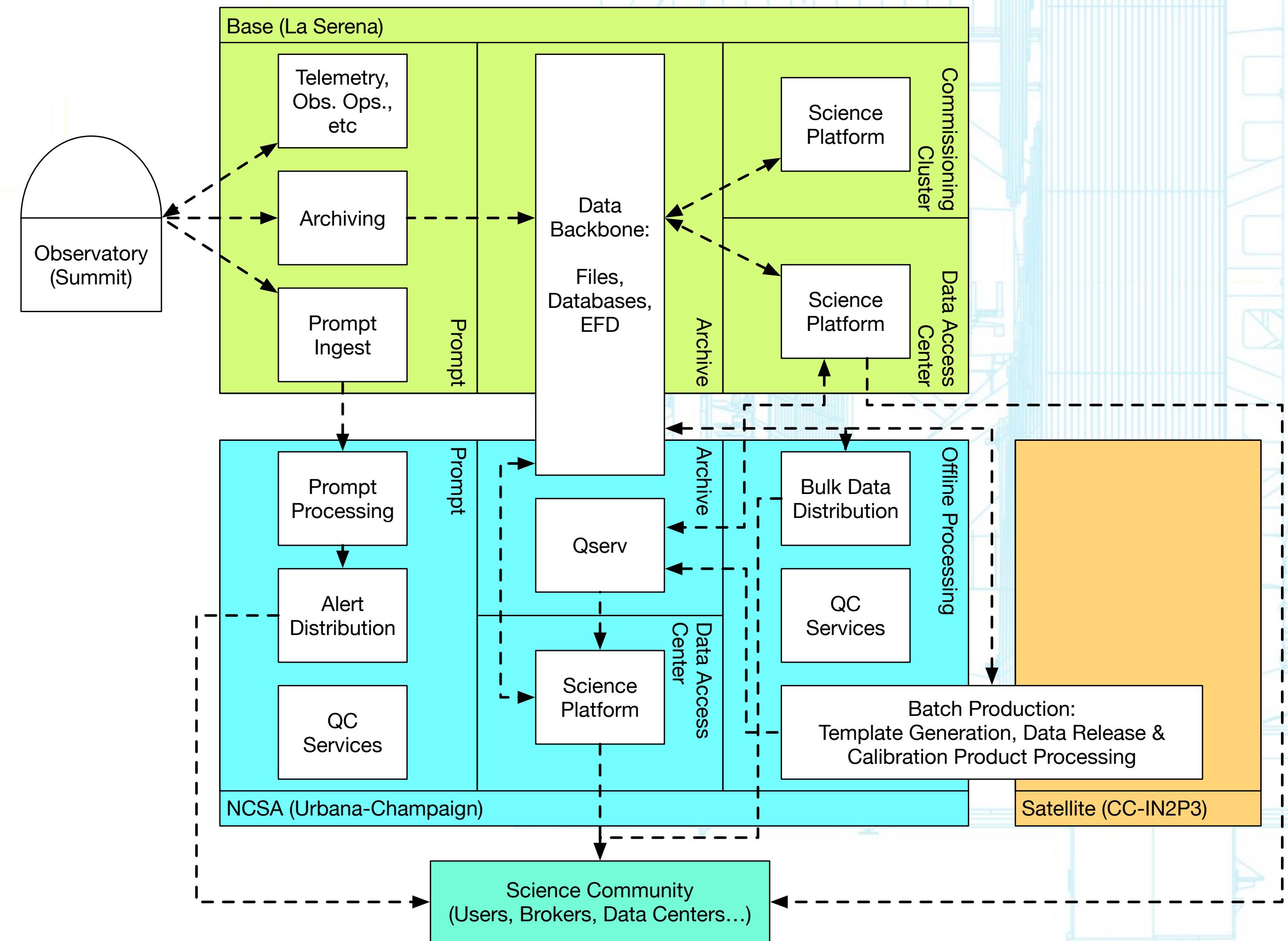
LSE-319 (Science Platform Vision), LDM-542 (Science Platform Design)



Data Management System



LDM-148 (DM System Design)

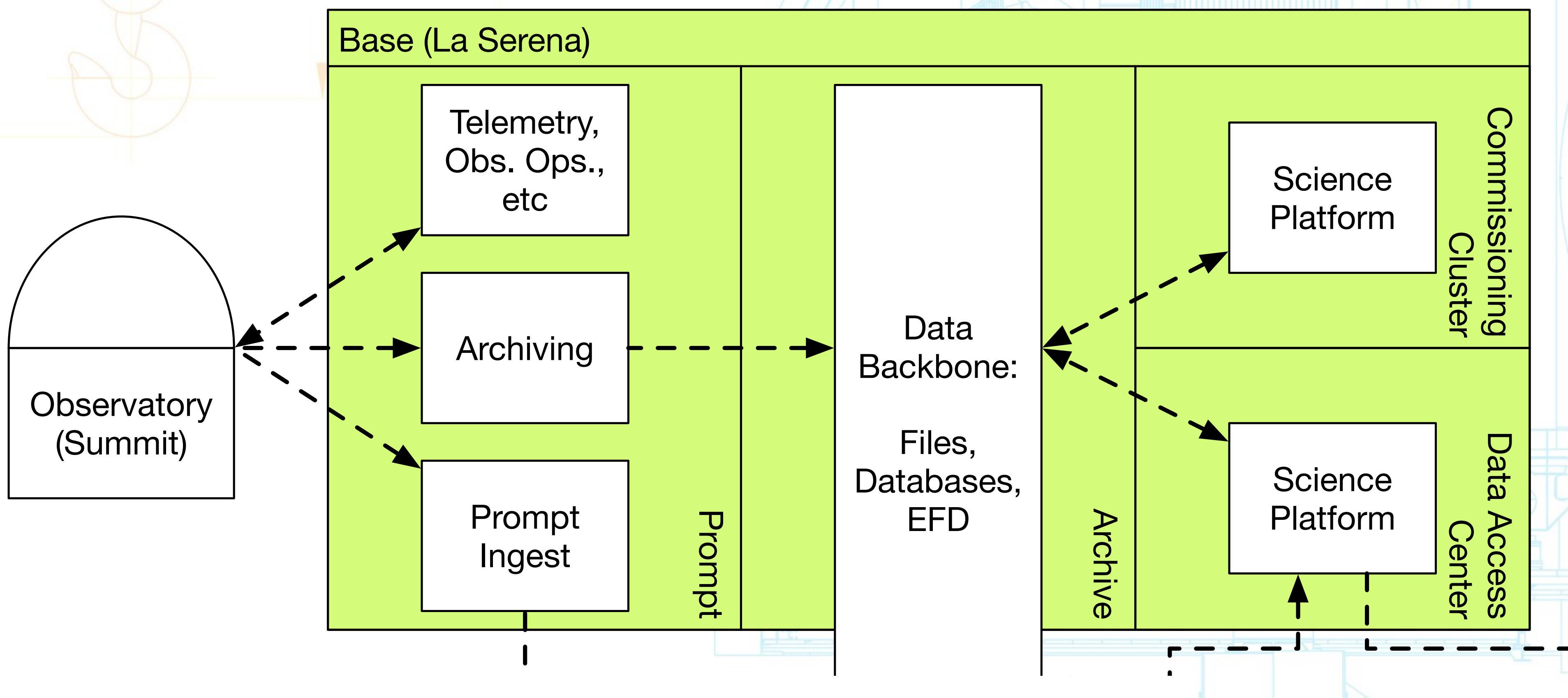


John Swinbank

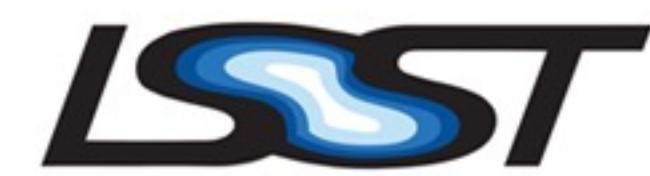
Data Management System



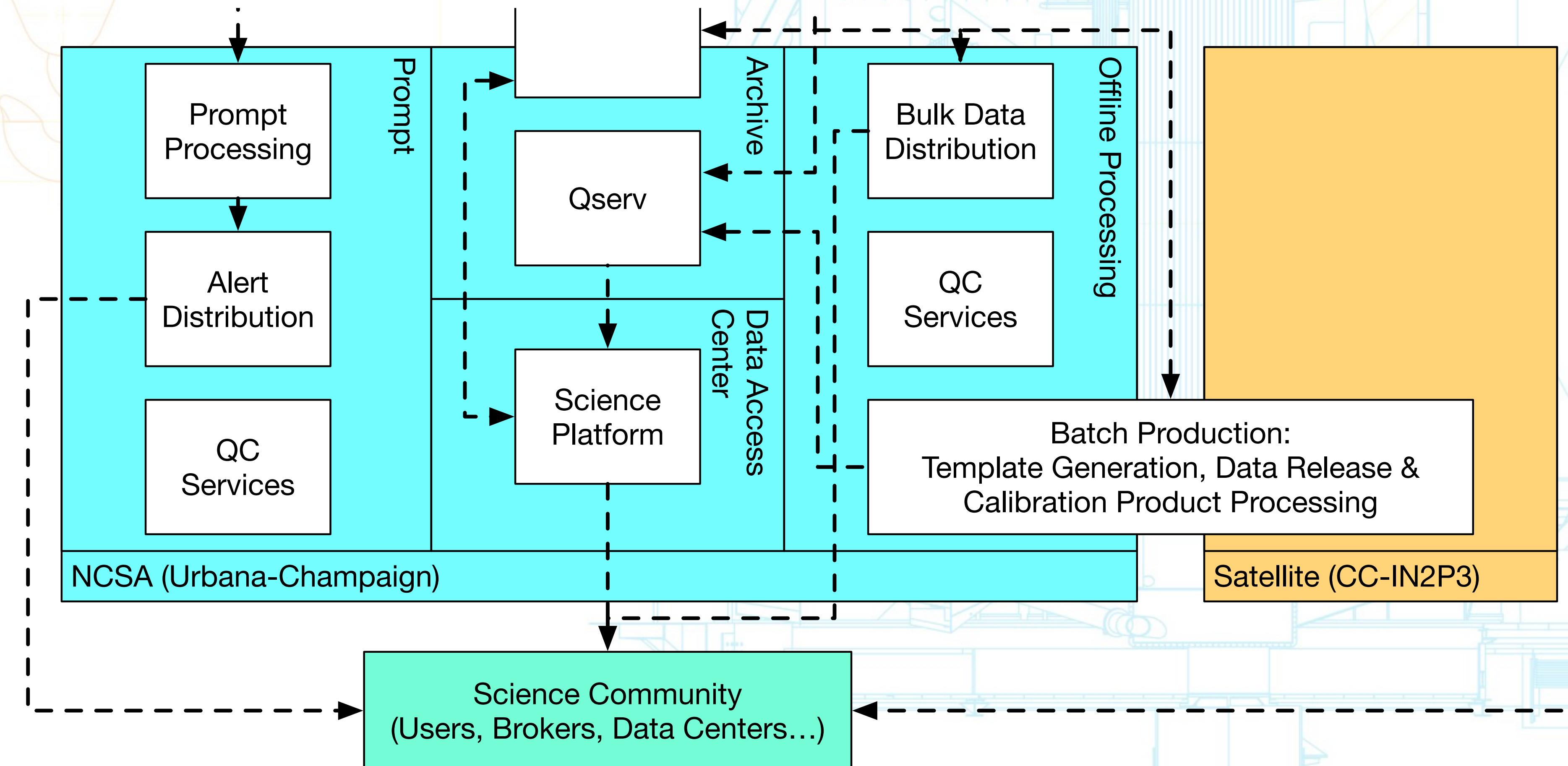
LDM-148



Data Management System



LDM-148

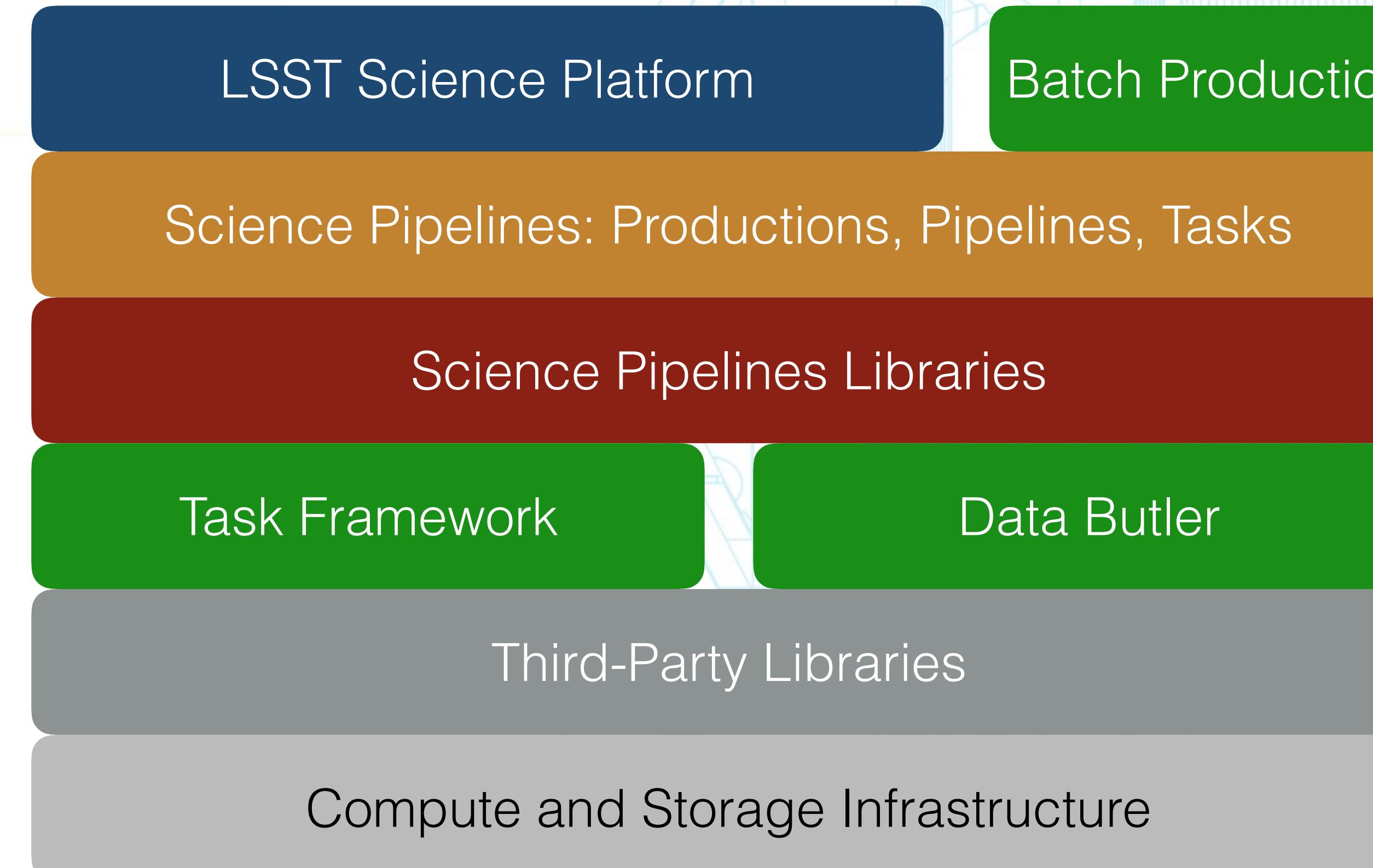


Software Layers



LDM-148

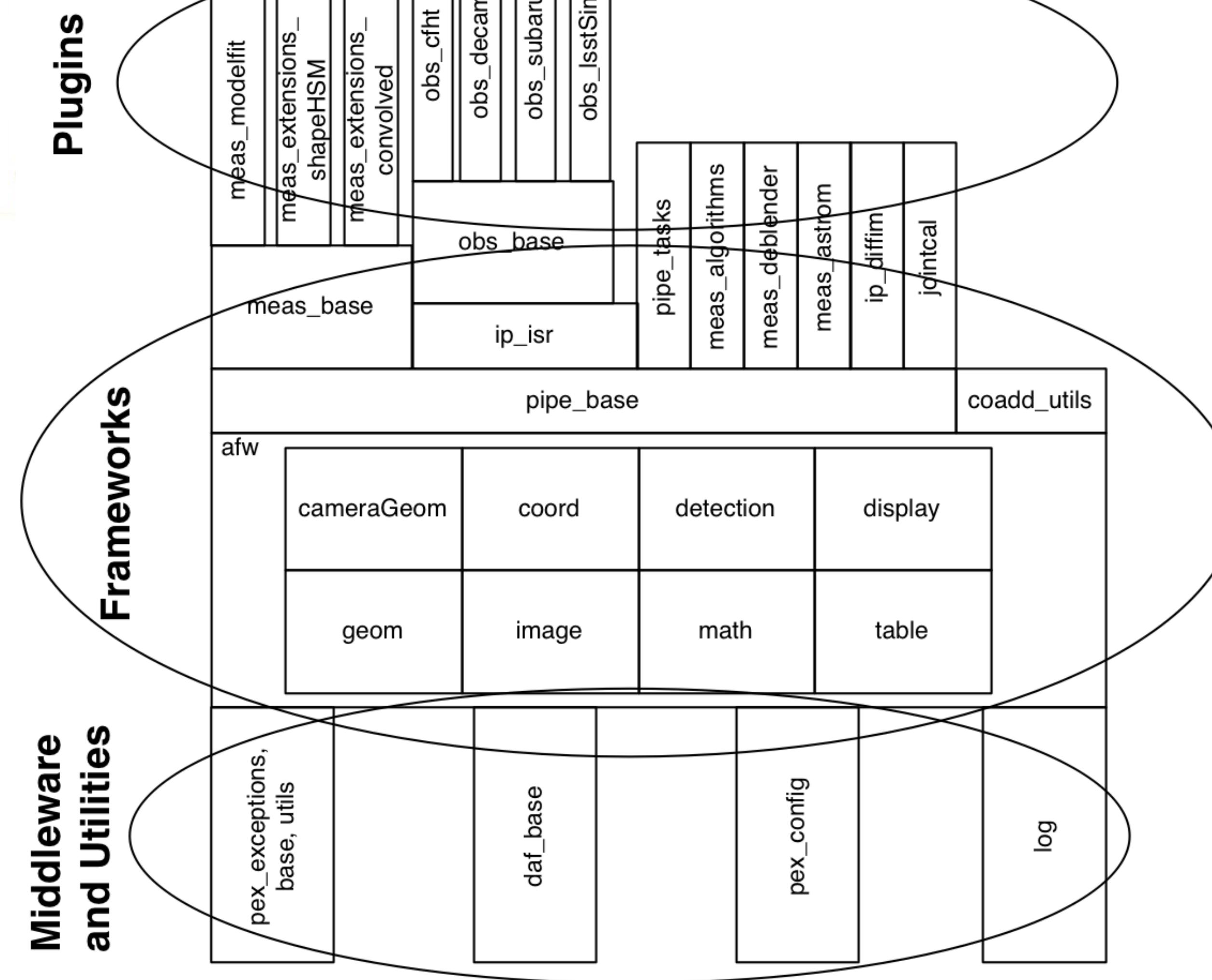
“Middleware”



"Stack" Details



LDM-148

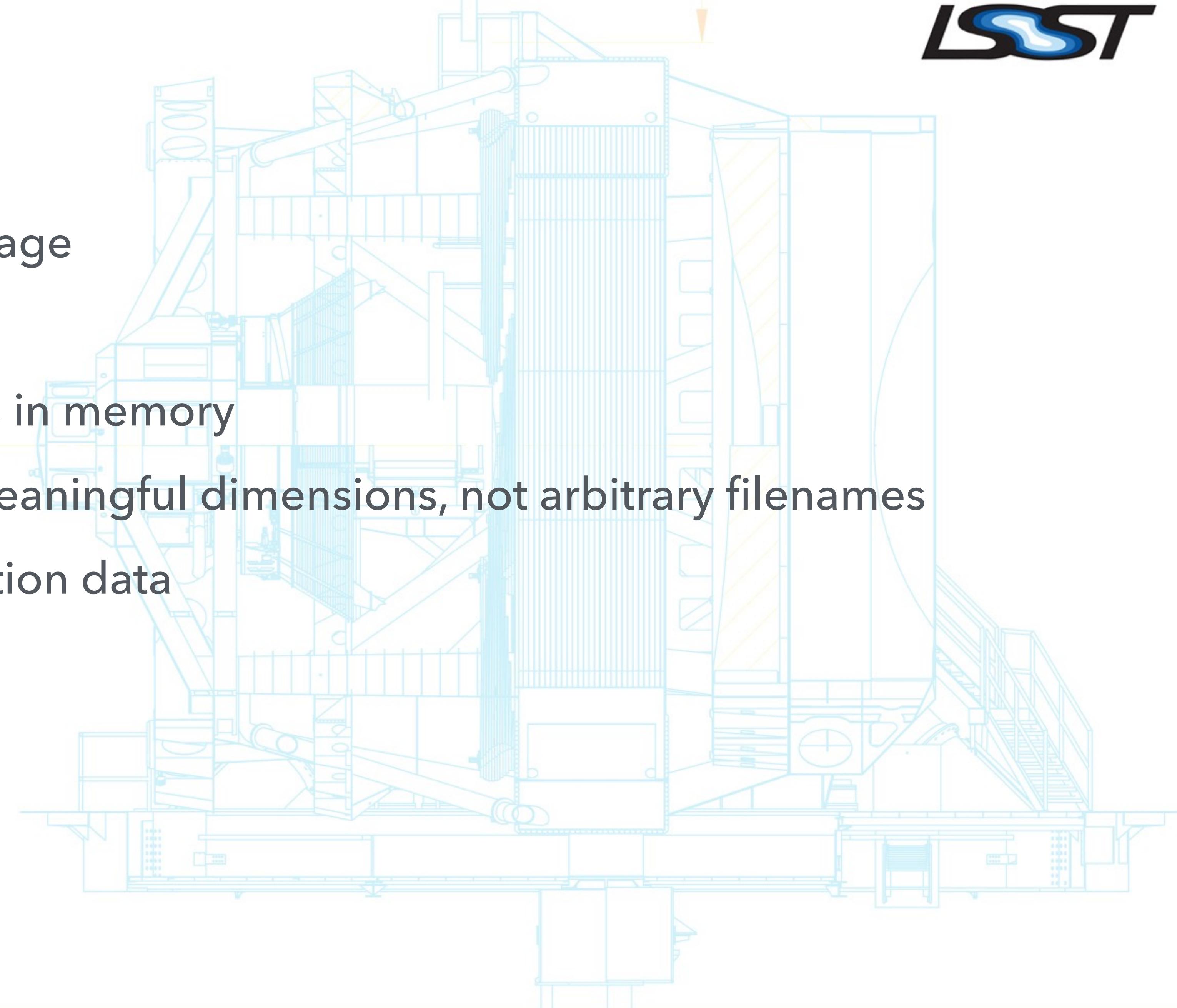


Why A Data Butler?



LDM-152 (Middleware Design)

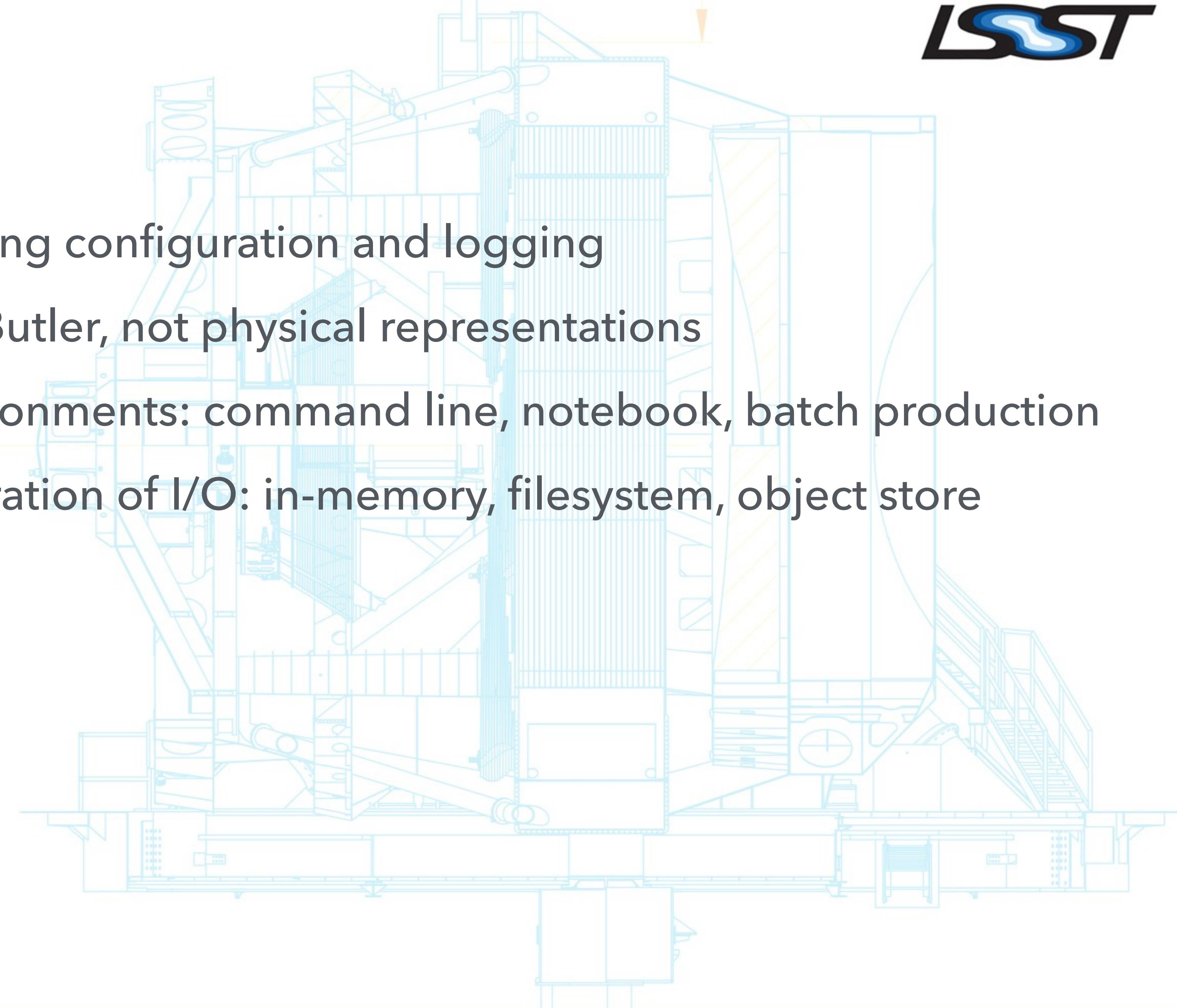
- Abstracts away underlying storage
- Abstracts away data format
- Delivers data as Python objects in memory
- Allows retrieval/selection by meaningful dimensions, not arbitrary filenames
- Allows “rendezvous” of calibration data



Why A Task?

LDM-152

- Standardizes interfaces, including configuration and logging
- Operates on objects from the Butler, not physical representations
- Allows execution in many environments: command line, notebook, batch production
- Allows separation and configuration of I/O: in-memory, filesystem, object store

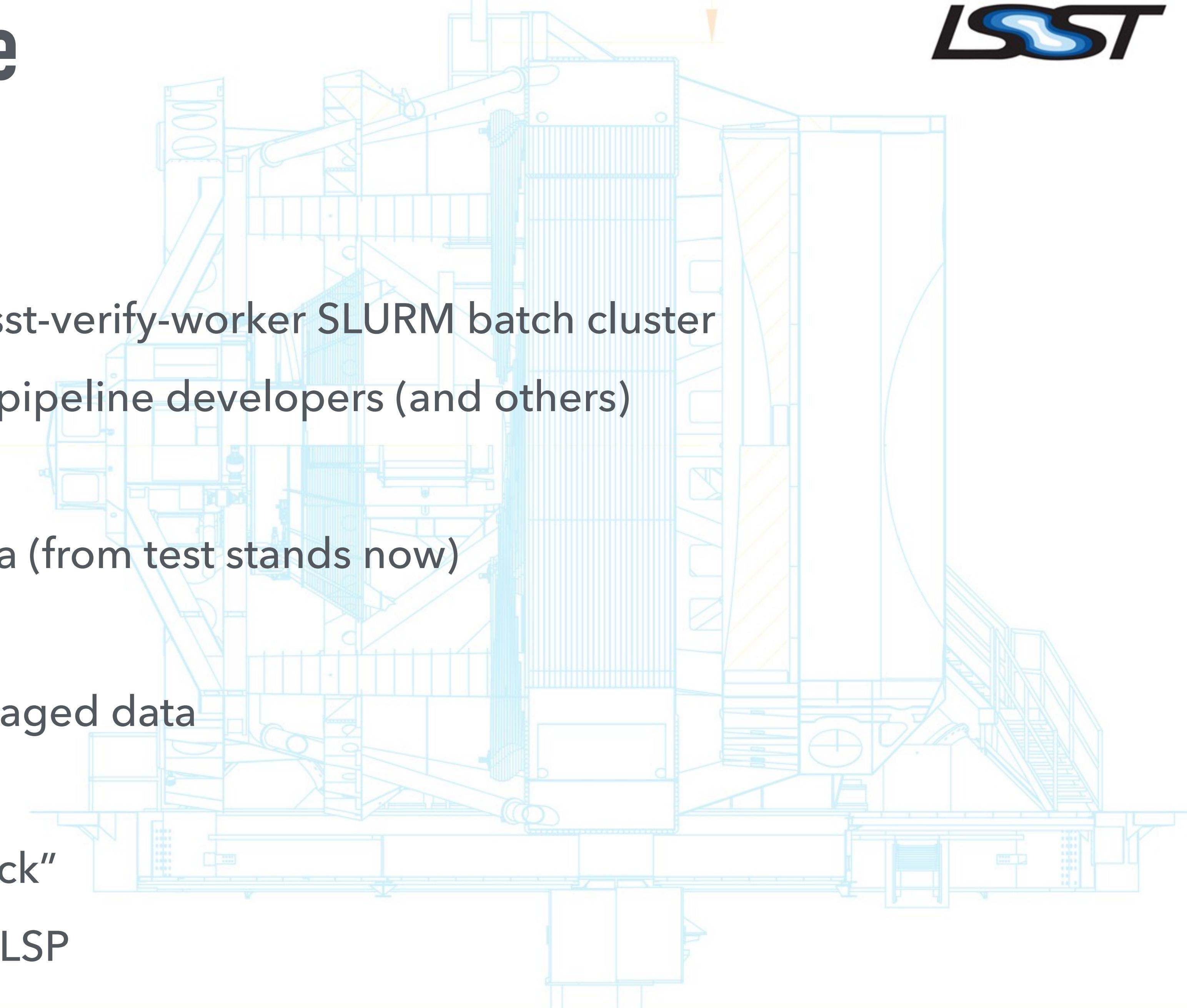


Developer Infrastructure



ncsa.illinois.edu

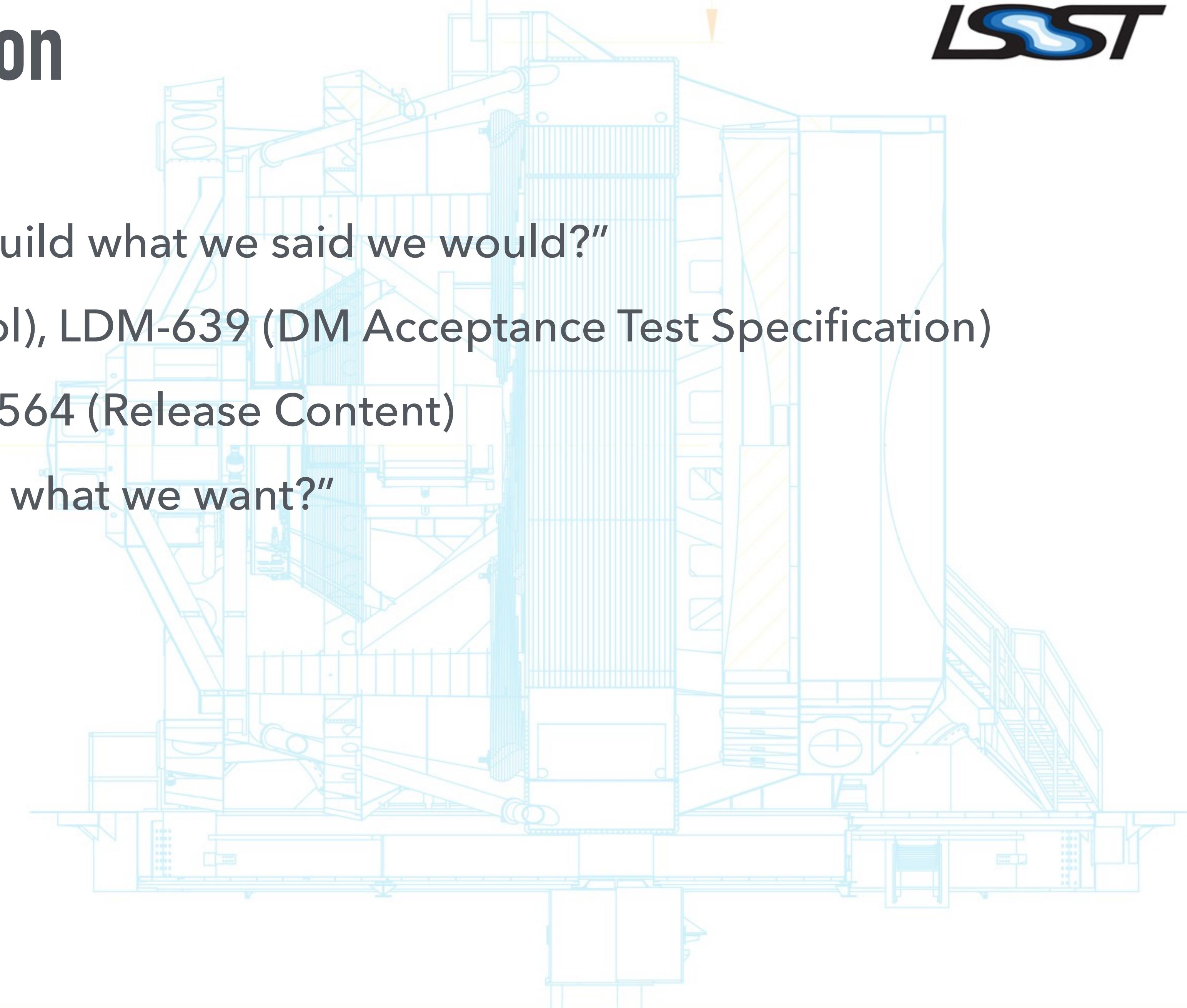
- Compute:
 - lsst-dev0[123] “head” nodes, lsst-verify-worker SLURM batch cluster
 - lsst-lsp-stable LSP instance for pipeline developers (and others)
- Storage:
 - /lsstdata for production data (from test stands now)
 - /datasets for precursor data
 - /project for shared, self-managed data
 - /scratch for temporary data
 - /software for the shared “stack”
 - ~/jhome for files visible in the LSP



Verification and Validation



- Verification answers “Did we build what we said we would?”
 - LDM-692 (Verification Control), LDM-639 (DM Acceptance Test Specification)
 - LDM-503 (Milestones), LDM-564 (Release Content)
- Validation answers “Does it do what we want?”
 - Goal of Commissioning



Commissioning and Operations Timeline



- Now:
 - LATISS in Tucson; near-operational services
 - ComCam in Tucson
 - LSSTCam at SLAC
- LATISS ships to Chile within weeks and delivers on-sky science data in 2020
- ComCam will follow in 2020, delivering mostly calibration data and possibly some science
- LSSTCam arrives early 2021, is on-sky by the end of that calendar year
- Operations phase starts October 2022

More Information

- Documentation Hub: <https://www.lsst.io>
 - Documents cited throughout this deck
 - <https://ls.st/> for documents and link shortening
- Developer Guide: <https://developer.lsst.io>
- Pipelines documentation: <https://pipelines.lsst.io>
- Doxygen C++ documentation: http://doxygen.lsst.codes/stack/doxygen/x_masterDoxyDoc/ or <https://ls.st/bpy>

