LARGE SYNOPTIC SURVEY TELESCOPE

Large Synoptic Survey Telescope (LSST)

Data Management Test Plan

William O'Mullane, John Swinbank, Mario Juric, Frossie Economou

LDM-503

Latest Revision: 2018-06-18

Abstract

This is the Test Plan for Data Management. In it we define terms associated with testing and further test specifications for specific items.

Change Record

Version	Date	Description	Owner name
	2017-01-13	First draft	William O'Mullane
1.0	2017-06-30	First approved release.	William O'Mullane
1.1	2017-07-04	Minor cleanups for review. Approved in RFC-358.	W. O'Mullane
1.2	2017-07-19	Milestone alignment with plan	W. O'Mullane
	2018-06-18	Milestone alignment with PMCS and derived from it	J. Swinbank
1.3	2018-06-18	Approved in RFC-492	J. Swinbank

Document source location: `https://github.com/lsst/LDM-503`

Contents

Data Management Test Plan

1 Introduction

This document provides an introduction to and overview of the approach to verification and validation which bas been adopted by the LSST Data Management Subsystem. Broadly, this approach consists of three aspects:

- *Verification* that the Data Management system as delivered meets the requirements placed upon it;
- *Validation* that the system as delivered meets the needs of the scientific community;
- *Rehearsing* the sustained operation of the system in operational scenarios.

This documentation describes how LSST Data Management is addressing each of these three requirements, and describes a series of high-level milestones and the accompanying test schedule. In addition, it briefly discusses the software development infrastructure that has been developed to support all three of these aspects of testing.

1.1 Objectives

We describe the test and verification approach for Data Management and describe various constraints and limitations in the testing to be performed. We also describe the program of rehearsals which will be undertaken to demonstrate the sustained operation of the Data Management system, and the validation exercises which will be performed on the partially and fully integrated system. We do not describe all tests in detail; those are described in dedicated test specifications for major components of Data Management.

1.2 Scope

This provides the approach and plan for all of Data Management. It covers interfaces between Data Management and components from other LSST subsystems but nothing outside of Data Management. This document is change-controlled by the DMCCB and will be updated in response to any requirements updates or changes of approach.

1.3 Assumptions

We will run large scale verification exercisess in order to demonstrate the system's end-to-end capability against its design specifications. A large amount of informal science verification and validation will be done in the the teams and documented in technical notes; in this test plan we are looking for verification of the broader system, demonstration of its *operability* — i.e. whether it can be run every day for the 10 year planned survey with a reasonable level of operational support – and to validate its capability to meet the scientific expectations of the community.

1.4 Applicable Documents

When applicable documents change a change may be required in this document.

LPM-55	LSST Quality Assurance Plan
LDM-148	DM Architecture
LDM-294	DM Project Management Plan
LDM-639	DM Acceptance Test Specification

1.5 References

[1] **[LDM-555]**, Becla, J., 2017, *Data Management Database Requirements*, LDM-555, URL `https://ls.st/LDM-555`

[2] **[LDM-533]**, Bellm, E.C., 2017, *Level 1 System Software Test Specification*, LDM-533, URL `https://ls.st/LDM-533`

[3] **[LDM-554]**, Ciardi, D., Dubois-Felsmann, G., 2017, *Science Platform Requirements*, LDM-554, URL `https://ls.st/LDM-554`

[4] **[LSE-79]**, Claver, C., The LSST Commissioning Planning Team, 2017, *System AI&T and Commissioning Plan*, LSE-79, URL `https://ls.st/LSE-79`

[5] **[LSE-29]**, Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2017, *LSST System Requirements (LSR)*, LSE-29, URL https://ls.st/LSE-29

[6] **[LSE-30]**, Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2018, *Observatory System Specifications (OSS)*, LSE-30, URL https://ls.st/LSE-30

[7] **[LSE-61]**, Dubois-Felsmann, G., Jenness, T., 2017, *LSST Data Management Subsystem Requirements*, LSE-61, URL https://ls.st/LSE-61

[8] **[LDM-639]**, Guy, L., 2018, *DM Acceptance Test Specification*, LDM-639, URL https://ls.st/LDM-639

[9] **[LPM-17]**, Ivezić, Ž., The LSST Science Collaboration, 2011, *LSST Science Requirements Document*, LPM-17, URL https://ls.st/LPM-17

[10] **[LDM-148]**, Lim, K.T., Bosch, J., Dubois-Felsmann, G., et al., 2017, *Data Management System Design*, LDM-148, URL https://ls.st/LDM-148

[11] **[LDM-552]**, Mueller, F., 2017, *Qserv Software Test Specification*, LDM-552, URL https://ls.st/LDM-552

[12] **[LDM-294]**, O'Mullane, W., Swinbank, J., Jurić, M., DMLT, 2017, *Data Management Organization and Management*, LDM-294, URL https://ls.st/LDM-294

[13] **[LPM-55]**, Sweeney, D., McKercher, R., 2013, *Project Quality Assurance Plan*, LPM-55, URL https://ls.st/LPM-55

[14] **[LDM-534]**, Swinbank, J.D., 2017, *Level 2 System Software Test Specification*, LDM-534, URL https://ls.st/LDM-534

[15] **[LDM-538]**, Unknown, 2017, *Base Enclave Test Specification*, LDM-538, URL https://ls.st/LDM-538

[16] **[LDM-540]**, Unknown, 2017, *LSST Science Platform Test Specification*, LDM-540, URL https://ls.st/LDM-540

1.6 Definitions, Acronyms, and Abbreviations

Acronym	Description
API	Application Programming Interface
CI	Continuous Integration

CPU	Central Processing Unit
CTIO	Cerro Tololo Inter-American Observatory
DAC	Data Access Center
DAX	Data Access Services
DBB	Data BackBone
DM	Data Management
DMCCB	DM Change Control Board
DRP	Data Release Production
EFD	Engineering Facilities Database
EPO	Education and Public Outreach
HSC	Hyper Suprime-Cam
ICD	Interface Control Document
ID	Identifier (Identification)
JIRA	issue tracking product (not an acronym, but a truncation of Gojira, the Japanese name for Godzilla)
KPM	Key Performance Metric
LSST	Large Synoptic Survey Telescope
LaTeX	(Leslie) Lamport TeX (document markup language and document preparation system)
NCSA	National Center for Supercomputing Applications
OCS	Observatory Control System
OSS	Operations Support System
PMCS	Project Management Control System
PSF	Point Spread Function
QA	Quality Assurance
QC	Quality Control
Qserv	Proprietary LSST Database system
RC	Reference Catalog
SPR	Software Problem Report
SQuaRE	Science Quality and Reliability Engineering
SST	System Science Team
SV	Science Validation
TB	TeraByte
TBD	To Be Defined (Determined)

UX	User interface widget
VC	Verification Cluster
VCD	Verification Control Document
WCS	World Coordinate System

2 Verification Tests

Our approach towards verifying the successful delivery of the Data Management System follows standard engineering practice.

We regard the system as being successfully completed when all of the high level requirements placed upon it, as defined in LSE-61 — the *Data Management System Requirements* — have been verified. The approach which will be taken to verifying each requirement is described in LDM-639, the *DM Acceptance Test Specification*. The test specifiation covers all aspects of the tests, as described in Section 2.3. Any given requirement may have multiple test cases associated with it in the specification, and these tests will be phased to account for incremental delivery depending on the need for certain functionality at a specific time.

In addition to the high level requirements on the overall Data Management system, lower-level requirements documents describe requirements placed upon specific parts of the system (for example, LDM-554 provides requirements on the LSST Science Platform, and LDM-555 on the DM database system). Each of these requirements documents is accompanied by a test specification (LDM-540 and LDM-552 in the case of the previous examples; see also Table 1), with the relationship between them being the same as for the high-level requirements.

In some cases, high-level test specifications may call out individual lower level specifications to demonstrate that some high-level requirement has been satisfied by the low-level component in isolation. In general, though, high-level tests are expected to demonstrate the successful integration and overall functionality of the entire DM system, not the proper operation of individual components within.

Although individual test cases may be executed at any time, it is anticipated that major testing campaigns will be undertaken to demonstrate the successful completion of major milestones in the Data Management construction effort. The schedule for these milestones is shown in

TABLE 1: Components of the Data Management system with the test specifications to verify them. A cyan background indicates that a test specification is currently available; yellow, that one is being drafted at time of writing; orange, that the existing test specification is under revision.

Component	Test Specification
NCSA Enclave	LDM-532
Level 1 System	LDM-533
Level 2 System	LDM-534
Data Backbone	LDM-535
Data Services	LDM-536
Data Backbone Infrastructure	LDM-537
Raw Image Archiving Service	LDM-538
Science Platform	LDM-540
Commisioning Cluster	LDM-541
Qserv	LDM-552
Data Management Acceptance	LDM-639

Section 6, while Section 7 provides further details as to the contents of each one. The mapping of LDM-639 test specifications to particular milestones is ongoing as of summer 2018.

2.1 Managing and Reporting Test Execution

As described above, requirements and test specifications are provided in baselined documents. These documents provide curated views on the Jira *LSST Verification and Validation* project which underlies the LSST-wide test effort. The Jira system provides "scripts" that testers will follow when carrying out tests, and tracks information about tesst execution and results. This information enables us to generate reports on the execution of each test[1], and ultimately to build a Verification Control Document (VCD; see Figure 1). The VCD will provide the verification status of each DM requirement (in terms of the fraction of test cases pertaining to that requirement which have been successfully executed).

2.2 Components Under Test

The components of the DM ystem are outlined in LDM-294 and described in detail in LDM-148; a summary is shown in Figure 2. The test specifications covering these components are

[1] These test reports may, on occasion, be issued as Word or LaTeX documents, but this is not, in general, required.

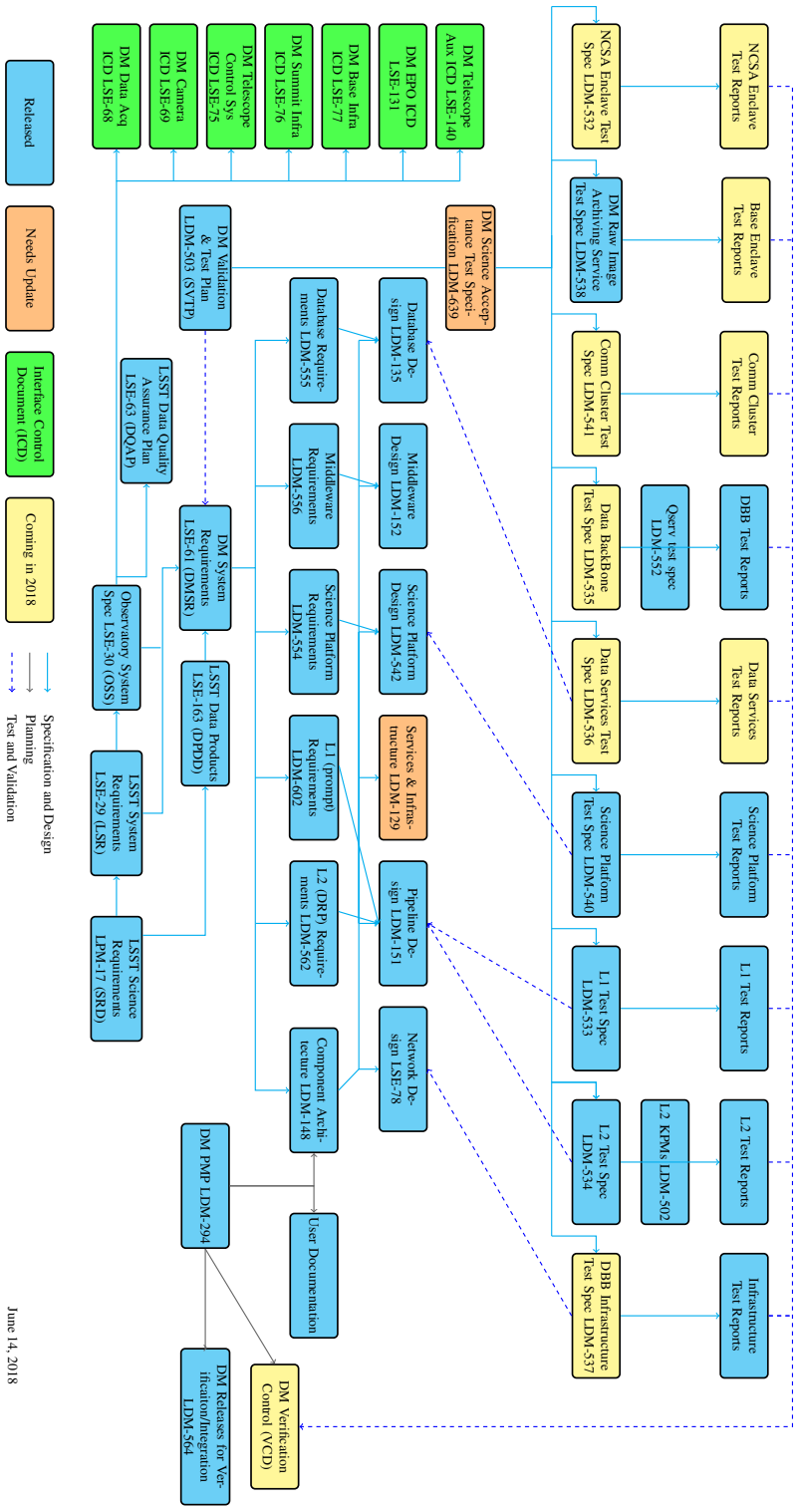

FIGURE 1: The Data Management document tree.

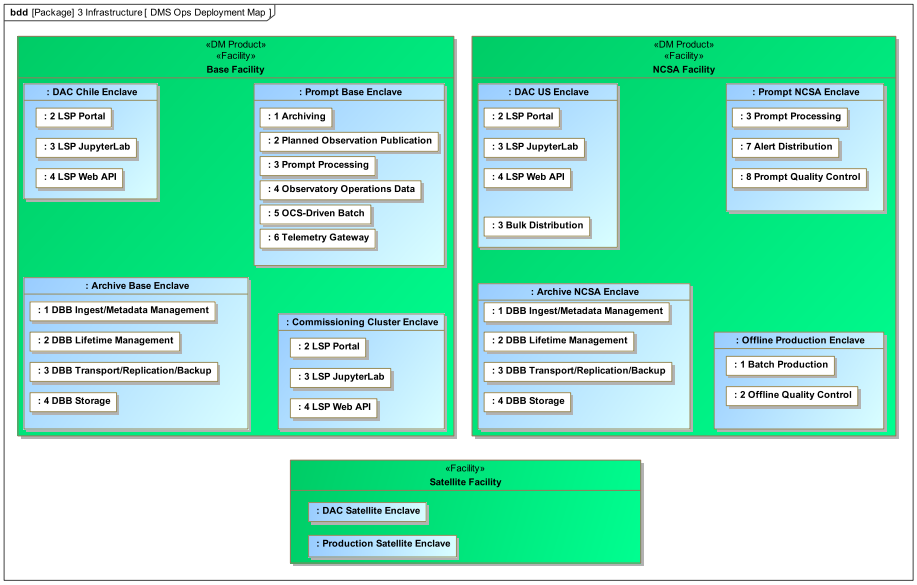

FIGURE 2: DM components as deployed during Operations. For details, refer to LDM-148.

shown in Table 1, but note that, at time of writing, the document tree is being refactored and document numbers are not currently available for all components. Based on those components we can see the set of Test Specifications needed in Table 1. At time of writing, document numbers are not available for all second-level components.

The test items covered in this test plan are:

- The Data Management System and its primary components for testing and integration purposes. These are listed in Table 1. All components listed in orange and yellow have specifications in the corresponding documents listed. Major sub-components in white may have individual test specifications or be addressed in the component they are under depending on applicable factors such as whether they are scheduled for testing at the same time and/or whether they share architectural components or are largely distinct.
- The external interfaces between Data Management and other sub-systems. These are described in DocuShare collection 5201.
- Operational procedures like Data Release Process, the Software Release Process and

the Security Plan.

2.3 Testing Specification Document Format

As described in Section 2.1, test specifications consist primarily of views on the test cases managed in the Jira "LSST Verification and Validation" project. The format of these test cases has been developed in conjunction with the LSST Systems Engineering Team. Each test case will include:

- A description of the environment in which the test case is to be carried out (e.g. hardware platform) and a description of how they differ from the operational system in tests prior to final integration (e.g. interfaces that may be mocked without affecting that component's testing).
- The inputs (such as data, API load, etc.) that are to be used in the test.
- Pass-fail criteria on any metrics or other measurements.
- How any outputs that are used to determine pass/fail (e.g. data or metrics) are to be published or otherwise made available.
- A software quality assurance manifest, listing (as relevant) code repositories, configuration information, release/distribution methods and applicable documentation (such as installation instructions, developer guide, user guide etc.)

In addiiton to the collection of test cases, the test specification will include:

- A list of components being tested within the scope of the test specification document.
- A list of features in those components that are being explicitly tested.
- The relationship between features under test and the identified requirements for the component.

2.4 Roles and Personnel

Each test specification must make clear who the *tester* is.

The tester is responsible for executing the test cases following the script provided in the Jira "LSST Verification and Validation" project (Section 2.3).

Testers submit details of test execution to Jira project, where it is used to log test execution and may be used to generate test reports. The information captured in Jira will also be used to populate the Verification Control Document (see Section 2).

Tests and procedures will sometimes fail: a test specification may be re-run several times until it passes, but testers will log an explanation than indicates that any failures were understood (e.g. they were due to a fault that was fixed) or repeated sufficient times to ensure that passing the test was not transient success. Issues which cannot be resolved by the tester in the course of carrying out the test willl be reported as "Software Problemn Reports" (SPRs) through the Data Management ticketing system (the Jira "Data Management" project at the time of this document revision). The DMCCB, or an individual designated by it, will be tasked with assessing the SPRs determining the timescale for re-executing the test procedure.

Other parties that have a relevant role in Data Management verification are identified in the appropriate sections of the document; these are involved in their primary capacity (e.g. the DM Systems Engineer) and so are not individually listed in this section.

2.5 Pass/Fail Criteria

A test case will be considered "Passed" when:

- All of the test steps of the Test Case are completed; and
- All open SPRs from this Test Case are considered noncritical by DMCCB.

A test case will be considered "Partially Passed" when:

- Only a subset of all of the test steps in the Test Case are completed and/or there remain open SPRs which are regarded as critical by the DMCCB; but
- The DMCCB regards overall purpose of the test as having been met.

A test case will be considered "Failed" when:

- Only a subset of all of the test steps in the Test Case are completed and/or there remain open SPRs which are regarded as critical by the DMCCB; and

- The DMCCB regards overall purpose of the test as not having been met.

Note that in LPM-17 science requirements are described as having a minimum specification, a design specification and a stretch goal. While we preserve these distinctions in some DM tooling for internal tracking purposes, for the purposes of these tests, it is the design specification that is verified as having been met for a test to pass without intervention of the DMCCB. Ultimately, if it proves impossible to satisfy a requirement at design specification, LSST Project level approval is required to accept the minimum specification.

2.6 Constraints and Limitations

2.6.1 Procedural and Technical Limitations

- The Data Management system must be verified before the complete LSST system can be completed. Verification is therefore carried out using precursor datasets [2], simulated data, and — where available — with engineering and pre-release data from the as yet incomplete LSST system.
- Metric measurements and operational rehearsals during construction may not involve critical operational systems that are still in development. For example, while computational performance is being measured, computationally dominant algorithmic steps such as deblending and multi-epoch fitting may only be modeled, since they have not yet been implemented; operational rehearsals are done without the factory LSST workflow system; etc.

2.6.2 Requirements Traceability Constraints

The Data Management verification plan is based entirely on requirements captured in the DM System Requirements (LSE-61). It does not refer to higher level requirements documentation, such as the LSST System Requirements (LSE-29) or the Observatory System Specifications (LSE-30); rather, we assume that all higher level requirements have been correctly flowed down to DM. In practice, the Systems Engineering team continues to refine the flow-down of higher level requirements and issue updates to LSE-61; this test plan must both anticipate and be responsive to those updates.

[2] e.g. from Hyper Suprime-Cam; Section 7.4

3 Operations Rehearsals

The operability of the Data Management system is demonstrated through a series of operations rehearsals. Like verification tests , these rehearsals correspond to high level DM milestones (Section 6), and involve carrying out a specific set of activities under controlled conditions. As such, many of the considerations described in Section 2 also apply to rehearsals. However, the aim of the rehearsal is not to verify that the performance of the Data Management system meets some requirement, but to verify that it can be integrated and operated successfully, and to demonstrate and validate operational procedures.

The current schedule calls for six rehearsals to be carried out to test different aspects of the system (for example, one rehearsal addresses nightly operations, and another the production and curation of a data release). At time of writing, the activities to be undertaken as part of each operations rehearsal are currently being detailed. This schedule and scope of each exercise will be designed to align with the LSST Commissioning Plan (LSE-79).

Operations rehearsals require an *Rehearsal Coordinator* to oversee the process. This is a distinct role from that of the testers (Section 2.4, since they are (by definition) carrying out their operational roles during the rehearsal. For example, the rehearsal may not be directed by the Operations Manager, since that person has a major role in the rehearsal. An individual not involved in the rehearsal itself will be identified to perform this function.

4 Science Validation

4.1 Definition

We define DM Science Validation as the process by which we assess the as-built Data Management system meets the needs of the scientific community and other identified stakeholders.

We assess the projected and realized scientific usability of the system by periodically exercising the integrated system in a way that goes beyond synthetic unit and integration tests and verification of piece-wise requirements as described in previous sections. In other words, we *attempt to use the system in ways we expect it to be used by the ultimate users of the system, scientists*. An example may be performing a mock science study on the results of processing of precursor data, or performing a mock science-like activity (e.g., interactive analysis of

time-domain datasets) on a partially stood-up service (e.g., the Notebook aspect of the LSST Science Platform). We record and analyze any issues encountered in such usage, and feed this information back to the DM Science and DM development teams.

Science Validation exercises are designed to close the design-build-verify loop, and enable one to measure the degree to which the requirements, designs, the as-built system, and future development plans continue to satisfy stakeholder needs. They also provide valuable feedback about modifications needed to ensure the delivery of a scientifically capable system. Ultimately, SV activities transfer into commissioning SV activities and provide training to the future members of the Commissioning team.

4.2 Schedule and Execution

4.2.1 Schedule

Unlike the verification and rehearsal activities, which correspond to high level milestones, validation activities are planned and prepared in a rolling wave fashion in parallel with development activities (on a 6-month cycle, or perhaps a year). The SV activities will typically be designed so as to exercise the capabilities of the system expected to be delivered at the end of a given development cycle. The Science Validation (SV) team guides the definition of goals of those activities, in close consultation with the DM Project Manager.

By their nature, SV activities will typically lag behind deliveries of the (sub)system being verified – ideally, they will commence immediately upon delivery. Preparatory SV activities (e.g., identification and acquisition of suitable datasets, identification of potential Science Collaboration resources to include on the activity, or development of activity-specific analysis codes) will commence as early as feasible. DM SV Scientist will coordinate the execution of all SV activities.

SV activities should aim to take no longer than two months to conclude, to enable rapid actionable feedback to DM Management and DM Subsystem Science.

4.2.2 Execution

Science Validation activities typically follow the successful execution of unit and integration test activities described in the previous sections, especially the larger "dress rehearsals" and "data challenges" as listed in Section 6 (Master Schedule).

Following successful service stand-up or data challenge execution (at integration and unit test level), the generated data products or integrated services are turned over to the SV team. The SV team performs additional tests and data analyses to exercise the integrated system and assess its quality relative to expectations for the current phase of construction. This assessment is fed back to DM Subsystem Science and Systems Engineering teams to inform them about the status and needed improvements to the system.

Beyond reporting on the results, the SV team examines the tests or procedures developed in this phase and identifies those that are good new metrics of system quality and could be run in an automated fashion. These are fed back to the development teams for productizing and incorporation into the automated QC systems.

4.3 Deliverables

Key deliverables of Science Validation activities are:

- Reports on the assessed capability of the Data Management System to satisfy stakeholder needs. The assessments shall take into account the expected maturity of the system being tested.
- Recommendations for improvements and changes, both in the quality of as-constructed systems (i.e., what needs to be built differently or better, to make it more consistent with the system vision), as well as the overall system vision (i.e., recommendations on where the vision may need to be modified to fully respond to stakeholder needs).
- Measurements of performance metrics that do not lend themselves to easy automation (e.g., science activities requiring human involvement, like visual classification, or UX tests).
- Identification of new performance metrics to be tracked, including potential deliveries of code to the DM Construction and I&T teams for inclusion in automated quality control

pipelines.

- Other deliverables as charged when chartering a particular SV exercise.

4.4 Organization and Resources

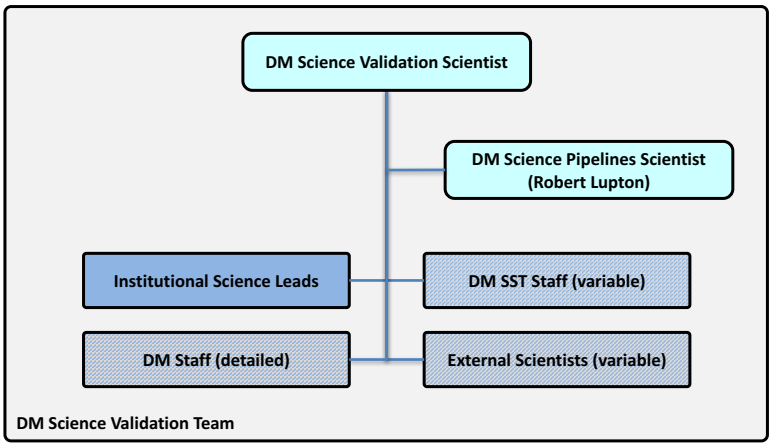

FIGURE 3: Organogram of the Data Management Science Validation Team. The group is chaired by the DM Science Validation Scientist, with the DM Science Pipelines Scientist and Institutional Science Leads making up the permanent membership. Depending on the SV activities being executed at any given time, the group may draw on additional temporary members from DM SST Staff, the broader DM Construction staff, as well as external scientists (e.g., Science Collaboration members committed to assisting SV goals). SV membership is reassessed on a cycle by cycle basis, with estimates incorporated in the long-term plan.

The DM Subsystem Scientist is accountable to the LSST Project Scientist for successful execution of DM Science Validation activities. This responsibility is delegated to the **DM Science Validation Scientist**, who leads the Science Validation (SV) team.

The SV team guides the definition of goals and receives the products of dress rehearsal activities, consistent with the long-term testing roadmap defined in Section 6. Decisions on strategic goals of SV exercises are made in close consultation and coordination with the DM Project Manager and Subsystem Scientist. The results of SV activities are reported to the DM Project Manager and Subsystem Scientist.

SV activities draw on resources of the DM System Science Team, but may also tap into the broader construction team if needed (and as jointly agreed upon with the DM Project Manager), as well as contributors from the LSST Science Collaborations. Additional members may

added as needed, depending on SV activities being considered and based on the recommendation of the DM SV Scientist and resource constraints.

The SV Scientist, the DM Science Pipelines Scientist, and all Institutional Science Leads are ex-officio members of the SV Team. DM Project Scientist and Managers are not formal members, but monitor the work of the group.

4.4.1 Example

An example of a Science Validation activity may be as follows:

- Based on the long-term development roadmap and new capabilities expected to be delivered, the at the beginning of a 6-month cycle the SV Team defines the goals of a data challenge to be executed at the end of the cycle. For the purposes of this example, we assume a major new feature to be delivered is astrometric calibration and estimation of proper motions.
- A small data release production using HSC data is defined that should result in a data set sufficient to measure the size and orientation of velocity ellipsoids in the Galactic halo. If such measurement are a success, they would independently validate the newly added global astrometric calibration and proper motion measurement capability.
- At the end the development cycle, the Science Pipelines team delivers to the proto-Operations team a documented and internally tested set of DRP pipelines with the new capabilities as defined above. The pipelines pass all unit and small-scale integration tests. The proto-Operations team deploys and re-verifies the received pipelines in the I&T environment designed to closely mimic the production environment. They verify that the pipeline integrates well with the orchestration system and is capable of executing medium-to-large scale processing. The pipelines pass integration tests.
- The data challenge is operationally planned and executed by the proto-Operations team, including the execution of any predefined QA metrics. The data products and test results are turned over to the Science Validation team.
- The Science Validation team performs the analysis needed to achieve SV exercise goals (the measurement of velocity ellipsoids, in this case).

- The results and conclusions derived from the data challenge are fed back to the DRP team, DM Project Management, and DM Subsystem Science; they may be used to assess the overall quality of the product, pass a formal requirement, and/or inform future construction decisions.
- Any newly developed but broadly useful tests are identified as such, and fed to the I&T team for inclusion into the battery of tests that are run on a regular basis.

5 Software Tools

All of the preceding aspects of testing — verification, science valdation, and operations rehearsals — rely on a robust and powerful software infrastructure. In this section, we briefly outline the tooling and practices that are in use within Data Management to ensure software quality and to verify that requirements are met. These tools are used continuously (e.g. to measure key performance metrics routinely) or periodically (e.g. software release characterizations).

5.1 Continuous Integration and Unit Testing

Per the Software Unit Test Policy described in the DM Developer guide, all DM code must be accompanied by an appropriate unit test suite. This suite is regularly executed by a continuous integration (CI) service[3], which is available for on-demand use by developers and for periodic testing (§§7.1 & 7.2). Irrespective of formally supported platforms, we have a practice of verifying that the codebase can run on at least two distinct operating systems/platforms as portability is often a good proxy for maintainability.

Roles and responsibilities in this area include:

- The DM Software Architect is responsible for setting the unit testing policy.
- The SQuaRE team is responsible for developing, operating and supporting continuous integration services.
- The SQuaRE team determines platform release practice in conjunction with the other teams, notably including Architecture.

[3] Currently based on Jenkins; `https://jenkins.io`

At the time of writing, we can calculate but do not formally track the code coverage of our unit test suite. A system to track this metric will be forthcoming.

5.2 Code Reviews

DM's process requires that every code change is subjected to peer review prior to being merged to the `master` branch. This is both as code quality verification and also to ensure that at least one other team-member has some familiarity with a particular part of the codebase. The code review process is described in the DM Developer Guide.

Roles and responsibilities in this area include:

- The DM Systems Engineering Team defines the development process and style guide including the code review standard.
- SQuaRE is responsible for supporting tooling to assist code review (e.g. linters, Jira-GitHub integration, etc).

5.3 Automated Requirements Verification and Metric calculation

DM uses a harness for continuous metric verification. In the software development context this is used for:

- Calculating performance metrics and alerting when they fail to meet requirements.
- Regression testing framework for any developer-supplied metric, with optional alerts when excursions occur from past values to verify that performance is not being degraded by new code or environments.
- Visualizing these results and linking them back to build and pull request information.
- Drill-down of those metrics in pre-defined visualization templates geared towards specific verification use-cases.

Roles and responsibilities in this area include:

- The pipeline teams are responsible for providing the code and data to demonstrate

compliance with requirements.

- SQuaRE is responsible for developing and operating the continuous metric verification services.
- Individual developers may contribute additional metrics as desired.

6 Master Schedule

The schedule for testing the system until operations commence (currently 2022) is outlined in Table 3. Tests named following the pattern "LDM-503-NN" (LDM-503-01, -02, etc) correspond to major [4] milestones of the DM project. These tests are closely tied to major integration events for the overall LSST system, as shown in Figure 4. This schedule also includes nightly and weekly tests which are executed throughout the construciton period by the continuous integration (CI) system (Section 5).

Table 3: List of High Level integration tests for DM

ID	Date	Location	Title
LDM-503-NLY	Nightly	CI System	Nightly integration tests
LDM-503-WLY	Weekly	CI System	Weekly integration tests
LDM-503-01	2017-11-30	NCSA	Science Platform with WISE data in PDAC
LDM-503-02	2017-11-30	NCSA	HSC reprocessing
LDM-503-03	2017-11-30	NCSA	Alert generation validation
LDM-503-04	2018-06-01	NCSA	Aux Tel DAQ integration functionality test
LDM-503-04b	2018-06-01	NCSA	Aux Tel DAQ interface Integration Verification and Spectrograph Operations Rehearsal
LDM-503-05	2018-07-02	NCSA	Alert distribution validation
LDM-503-08b	2018-09-06	NCSA	Small Scale CCOB Data Access
LDM-503-06	2018-11-30	NCSA	DM ComCam interface verification readiness
LDM-503-07	2018-11-30	NCSA	Camera data processing
LDM-503-08	2018-11-30	NCSA	Spectrograph data acquisition
LDM-503-09	2018-11-30	NCSA	Ops rehearsal for commissioning #1
LDM-503-09a	2018-11-30	NCSA	Pipelines Release Fall 2018
LDM-503-10	2018-11-30	NCSA	DAQ validation
LDM-503-10b	2019-08-06	NCSA	Large Scale CCOB Data Access

[4] Level 2, in the parlance of LDM-294.

LDM-503-11a	2019-10-21	NCSA	ComCam Ops Readiness
LDM-503-11b	2019-10-21	NCSA	Pipelines Release Fall 2019
LDM-503-11	2019-10-31	NCSA	Ops rehearsal for commissioning #2
LDM-503-12	2019-10-31	NCSA	Ops rehearsal for commissioning #3
LDM-503-13a	2020-10-30	NCSA	Pipelines Release Fall 2020
LDM-503-13	2020-11-30	NCSA	Ops rehearsal for data release processing #1 (ComCam data)
LDM-503-14	2020-11-30	NCSA	DM Readiness for Science Verification
LDM-503-15a	2021-10-28	NCSA	Pipelines Release Fall 2021
LDM-503-15	2021-11-29	NCSA	Ops rehearsal for data release processing #2
LDM-503-16	2022-02-28	NCSA	Ops rehearsal for data release processing #3
LDM-503-17a	2022-08-31	NCSA	Final Pipelines Delivery
LDM-503-17	2022-09-30	NCSA	Final operations rehearsal

7 Major DM Milestones

7.1 Nightly software integration tests and releases (LDM-503-NLY)

Nightly throughout construction, the DM continuous integration system (Section 5) performs a complete rebuild of all major components of the Data Management codebase, executes their associated unit tests, and runs a series of automated small-scale integration tests. Failures are logged, and the DM team notified. The resulting build artefacts are packaged for release as a “nightly build”.

7.2 Weekly software integration tests and releases (LDM-503-WLY)

Nightly throughout construction, the DM continuous integration system (Section 5) performs a complete rebuild of all major components of the Data Management codebase, executes their associated unit tests, and runs a series of automated small-scale integration tests. Failures are logged, and the DM team notified. The resulting build artefacts are packaged for release as “weekly build”, and automatically made available for developer use on shared hardware.

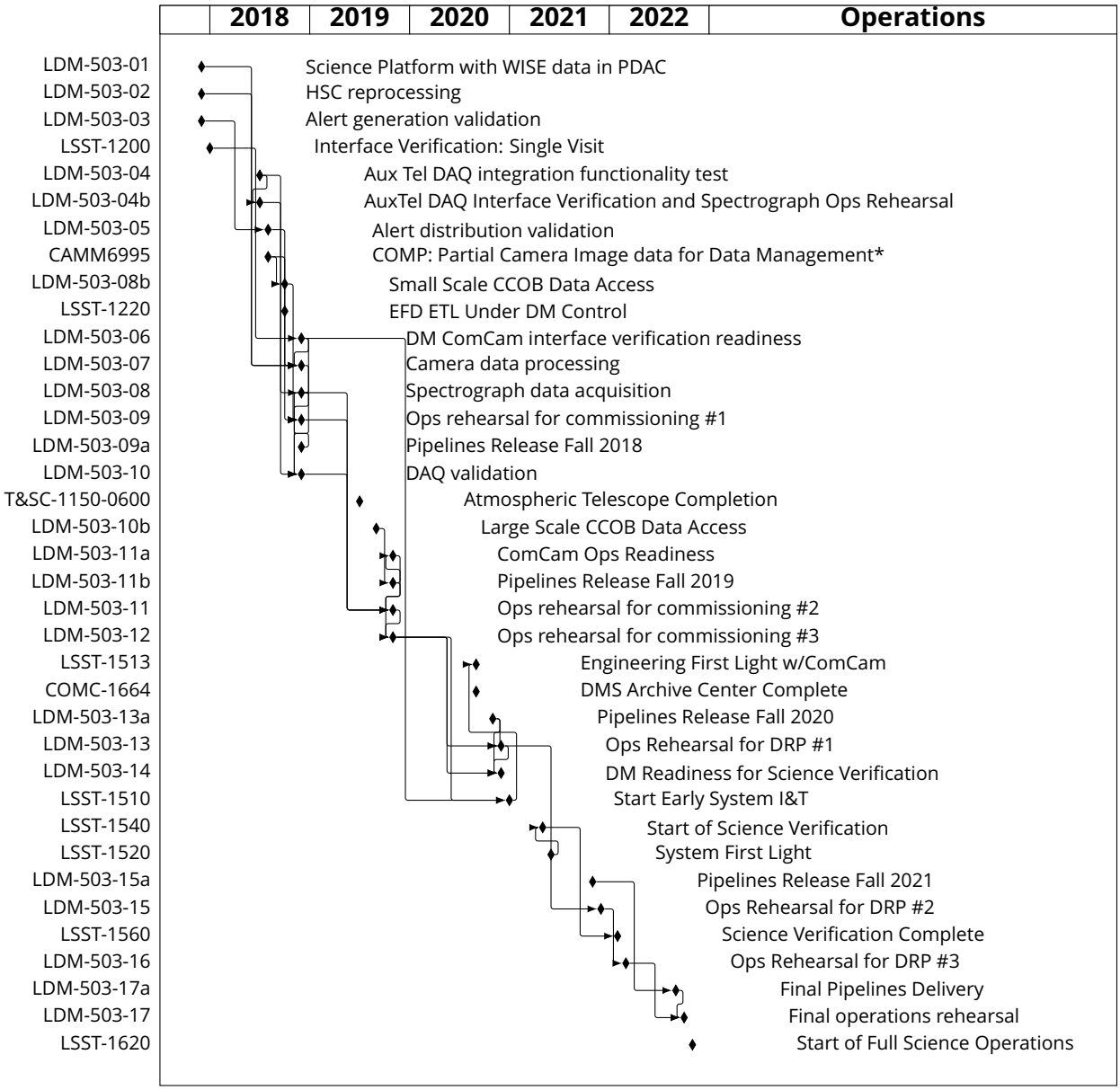

FIGURE 4: DM major milestones (LDM-503-x) in the LSST schedule.

7.3 Science Platform with WISE data in PDAC (LDM-503-01)

7.3.1 Specification

This test will be executed following the procedure defined in LDM-540 v1.0 §3.1 (LSP-00).

7.3.2 Description

SUIT continues PDAC development, adding the WISE data, further exercising the DAX dbserv and imgserv APIs, and taking advantage of metaserv once it becomes available

7.3.3 Comments

From DAX: need to be clear about which WISE datasets are to be loaded: the data wrangling effort required to download, inspect, convert, partition, and load each additional dataset is cumulatively non-trivial for DAX

7.4 HSC reprocessing (LDM-503-02)

7.4.1 Specification

This test will be executed following the procedure defined in LDM-534 v3.1 §3.1 (DRP-00).

7.4.2 Description

Validate the data products with the LSST stack match or improve upon HSC products. Validate the ops platform in NCSA, including installing the stack, starting & stopping production. Generate a validation data set for weekly integration and other tests.

7.5 Alert generation validation (LDM-503-03)

7.5.1 Specification

This test will be executed following the procedure defined in LDM-533 v1.1 §3.1 (AG-00).

7.5.2 Description

Validate the alert generation stack performance on several DECam & HSC datasets.

7.5.3 Comments

Validate the alert generation stack performance on several DECam and HSC datasets. "Stack" is probably ill-defined here: is this simply testing science logic, or are we going after a wider integration exercise?

7.6 Aux Tel DAQ integration functionality test (LDM-503-04)

7.6.1 Specification

This test will be executed following the procedure defined in LDM-538 draft version 2018-06-04[5], §3.1 (RAS-00).

7.6.2 Description

Demonstrate the writing of a well-formed raw image to the permanent record of the survey and provide rapid access for Observatory Operations staff.

7.7 Aux Tel DAQ interface Integration Verification and Spectrograph Operations Rehearsal (LDM-503-04b)

7.7.1 Specification

The execution procedure for this test is currently unspecified.

7.7.2 Description

The production Aux Tel data acquisition hardware should be available in Tucson. We will demonstrate integration with the adjacent archive systems.

[5] Currently undergoing review on RFC-488.

7.7.3 Comments

A minimal system that can archive simulated images from the Aux Tel DAQ and demonstrate that they can be retrieved.

7.8 Alert distribution validation (LDM-503-05)

7.8.1 Specification

The execution procedure for this test is currently unspecified.

7.8.2 Description

Validate alert distribution system and mini-broker fed by live or simulated live data.

7.8.3 Comments

Can we test a SUIT interface to the broker at this point? I believe it's not scheduled until later in construction.

7.9 Small Scale CCOB Data Access (LDM-503-08b)

7.9.1 Specification

The execution procedure for this test is currently unspecified.

7.9.2 Description

Demonstrate the ability to read data from the CCOB with 9 rafts, store it at the Data Facility and make it available on some cluster for analysis.

7.9.3 Comments

This was added as part of DM-13073 to show we can process optical bench data specifically CCOB 9 raft setup.

7.10 DM ComCam interface verification readiness (LDM-503-06)

7.10.1 Specification

The execution procedure for this test is currently unspecified.

7.10.2 Description

Test the operation of the DM system prior to ComCam availability in Tucson.

7.10.3 Comments

"The DM system" should use some further definition: what do we want to test here? Data flow from ComCam to the Data Backbone, or science processing of ComCam data? Note the LSE-79 requirements for DM services in support of ComCam in table 8. They're required by Nov 2019/Feb 2020; it may be more appropriate to test some of them at a later date?

7.11 Camera data processing (LDM-503-07)

7.11.1 Specification

The execution procedure for this test is currently unspecified.

7.11.2 Description

Partial camera data should be available to DM July 31st. We plan to test DM stack with it.

7.12 Spectrograph data acquisition (LDM-503-08)

7.12.1 Specification

The execution procedure for this test is currently unspecified.

7.12.2 Description

Demonstrate that we can acquire (and process?) data from the Spectrograph.

7.12.3 Comments

Per LSE-79, AuxTel delivery in Nov 2017 (ie, a year before this milestone) includes: EFD ETL service, Aux Telescope Archiving Service, Data backbone in support of Aux Telescope archiving. Do we need to schedule another test to cover that?

7.13 Ops rehearsal for commissioning #1 (LDM-503-09)

7.13.1 Specification

The execution procedure for this test is currently unspecified.

7.13.2 Description

Test how the system will run during commissioning.

7.13.3 Comments

Focus on ISR: we should test whatever we have available. See LSE-79 for a list of requirements.

7.14 Pipelines Release Fall 2018 (LDM-503-09a)

7.14.1 Specification

The execution procedure for this test is currently unspecified.

7.14.2 Description

Scientific pipelines release in support of operations rehearsal LDM-503-09. Accompanied by characterization report.

7.15 DAQ validation (LDM-503-10)

7.15.1 Specification

The execution procedure for this test is currently unspecified.

7.15.2 Description

There is a project milestone that DAQ/DM/Networks are available late 2019. We will run a full scale data acquisition test in to show this is ready.

7.16 Large Scale CCOB Data Access (LDM-503-10b)

7.16.1 Specification

The execution procedure for this test is currently unspecified.

7.16.2 Description

Demonstrate the ability to read data from the CCOB with 21 rafts, store at the Data Facility and make it available on some cluster for analysis.

7.16.3 Comments

This was added as part of DM-13073 to show we can process optical bench data specifically CCOB 21 raft setup. This extends LDM-503-8b.

7.17 ComCam Ops Readiness (LDM-503-11a)

7.17.1 Specification

The execution procedure for this test is currently unspecified.

7.17.2 Description

ComCam will be in use in Nov. The DM system must be ready to deal with it.

7.17.3 Comments

"The DM system" should use some further definition: what do we want to test here? Data flow from ComCam to the Data Backbone, or science processing of ComCam data? Note the LSE-79 requirements for DM services in support of ComCam in table 8. They're required by Nov 2019/Feb 2020; it may be more appropriate to test some of them at a later date?

7.18 Pipelines Release Fall 2019 (LDM-503-11b)

7.18.1 Specification

The execution procedure for this test is currently unspecified.

7.18.2 Description

Scientific pipelines release in support of operations rehearsal LDM-503-11. Accompanied by characterization report.

7.19 Ops rehearsal for commissioning #2 (LDM-503-11)

7.19.1 Specification

The execution procedure for this test is currently unspecified.

7.19.2 Description

More complete commissioning rehearsal: how do the scientists look at data? How do they provide feedback to the telescope? How do we create/update calibrations?

7.20 Ops rehearsal for commissioning #3 (LDM-503-12)

7.20.1 Specification

The execution procedure for this test is currently unspecified.

7.20.2 Description

Dress rehearsal: commissioning starts in April so by this stage we should be ready to do everything needed.

7.21 Pipelines Release Fall 2020 (LDM-503-13a)

7.21.1 Specification

The execution procedure for this test is currently unspecified.

7.21.2 Description

Scientific pipelines release in support of operations rehearsal LDM-503-13. Accompanied by characterization report.

7.22 Ops rehearsal for data release processing #1 (ComCam data) (LDM-503-13)

7.22.1 Specification

The execution procedure for this test is currently unspecified.

7.22.2 Description

ComCam data will now be available. Demonstrate its use in producing a data release.

7.22.3 Comments

Note that LSE-79 requires a suite of DM services in support of the full camera in May 2020. It seems inappropriate to test them as part of a commissioning ops rehearsal, but they are well before this data. Do we need another test milestone?

7.23 DM Readiness for Science Verification (LDM-503-14)

7.23.1 Specification

The execution procedure for this test is currently unspecified.

7.23.2 Description

Science Verification starts in April. Demonstrate that all required DM software is available.

7.23.3 Comments

SV will include calculating all KPMs to demonstrate that we are reaching the science requirements. That obviously means we'll need code which is both capable of reaching those requirements, and calculating the KPMs.

7.24 Pipelines Release Fall 2021 (LDM-503-15a)

7.24.1 Specification

The execution procedure for this test is currently unspecified.

7.24.2 Description

Scientific pipelines release in support of operations rehearsal LDM-503-15. Accompanied by characterization report.

7.25 Ops rehearsal for data release processing #2 (LDM-503-15)

7.25.1 Specification

The execution procedure for this test is currently unspecified.

7.25.2 Description

Science Verification data will now be available. Demonstrate its use in producing a data release.

7.26 Ops rehearsal for data release processing #3 (LDM-503-16)

7.26.1 Specification

The execution procedure for this test is currently unspecified.

7.26.2 Description

Test readiness for operations.

7.27 Final Pipelines Delivery (LDM-503-17a)

7.27.1 Specification

The execution procedure for this test is currently unspecified.

7.27.2 Description

Final scientific pipelines release from the DM Construction Project. Accompanied by characterization report.

7.28 Final operations rehearsal (LDM-503-17)

7.28.1 Specification

The execution procedure for this test is currently unspecified.

7.28.2 Description

Confirm readiness for operations.