



LARGE SYNOPTIC SURVEY TELESCOPE

# Large Synoptic Survey Telescope (LSST) Data Management Test Plan

**William O'Mullane, Mario Juric, Frossie Economou**

**LDM-503**

**Latest Revision: 2017-07-19**

This LSST document has been approved as a Content-Controlled Document by the LSST DM Change Control Board. If this document is changed or superseded, the new document will retain the Handle designation shown above. The control is on the most recent digital document with this Handle in the LSST digital archive and not printed versions. Additional information may be found in the corresponding DM RFC.

## **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

*Document source location:* <https://github.com/lsst/LDM-503>



# Contents

- 1 Introduction 1**
  - 1.1 Objectives . . . . . 1
  - 1.2 Scope . . . . . 1
  - 1.3 Assumptions . . . . . 1
  - 1.4 Applicable Documents . . . . . 2
  - 1.5 References . . . . . 2
  - 1.6 Definitions, Acronyms, and Abbreviations . . . . . 3
  
- 2 Roles and Reporting 4**
  
- 3 DM Verification Approach 5**
  - 3.1 Reports . . . . . 5
  - 3.2 Components Under Test . . . . . 5
  - 3.3 Testing Specification Document Format . . . . . 9
  
- 4 Pass/Fail Criteria 10**
  
- 5 Constraints and Limitations 10**
  - 5.1 Procedural and Technical Limitations . . . . . 11
  - 5.2 Requirements Traceability Constraints . . . . . 11
    - 5.2.1 Scientific . . . . . 11
    - 5.2.2 Computational . . . . . 11
    - 5.2.3 KPMs . . . . . 12
  - 5.3 Interfaces . . . . . 12
  
- 6 Master Schedule 13**
  
- 7 Verification Tests 15**
  - 7.1 Science Platform with WISE data in PDAC (**LDM-503-1**) . . . . . 15
  - 7.2 HSC reprocessing (**LDM-503-2**) . . . . . 17
    - 7.2.1 Personnel . . . . . 17



7.2.2	Open issues . . . . .	17
7.2.3	Datasets . . . . .	17
7.2.4	Calibration Products Production . . . . .	19
7.3	Alert generation validation ( <b>LDM-503-3</b> ) . . . . .	20
7.4	Aux Tel DAQ integration functionality test ( <b>LDM-503-4</b> ) . . . . .	21
7.5	Test Report: Aux Tel DAQ interface Integration Verification and Spectrograph Operations Rehearsal ( <b>LDM-503-4b</b> ) . . . . .	21
7.6	Alert distribution validation ( <b>LDM-503-5</b> ) . . . . .	21
7.7	DM ComCam interface verification readiness ( <b>LDM-503-6</b> ) . . . . .	21
7.8	Camera data processing ( <b>LDM-503-7</b> ) . . . . .	21
7.9	Spectrograph data acquisition ( <b>LDM-503-8</b> ) . . . . .	22
7.10	Verification tests in advance of pre-ops rehearsal for commissioning #1 ( <b>LDM- 503-9</b> ) . . . . .	22
7.11	DAQ validation ( <b>LDM-503-10</b> ) . . . . .	22
7.12	DM ComCam operations readiness ( <b>LDM-503-11a</b> ) . . . . .	22
7.13	Verification tests in advance of pre-ops rehearsal for commissioning #2 ( <b>LDM- 503-11</b> ) . . . . .	22
7.14	Verification tests in advance of pre-ops rehearsal for commissioning #3 ( <b>LDM- 503-12</b> ) . . . . .	23
7.15	Ops rehearsal DRP (ComCam data) ( <b>LDM-503-13</b> ) . . . . .	23
7.16	DM Software for Science Verification ( <b>LDM-503-14</b> ) . . . . .	23
7.17	Verification tests in advance of Full Operations rehearsal ( <b>LDM-503-15</b> ) . . . . .	23
7.18	Operations Rehearsal for Data Release Production ( <b>LDM-503-16</b> ) . . . . .	23
7.19	Verification tests for full DRP ( <b>LDM-503-17</b> ) . . . . .	24
<b>8</b>	<b>Software Tools</b>	<b>24</b>
8.1	Continuous Integration and Unit Testing . . . . .	24
8.2	Code Reviews . . . . .	25
8.3	Automated Requirements Verification and KPM Measurement . . . . .	25
<b>9</b>	<b>Operations Validation</b>	<b>26</b>



<b>10 Science Validation</b>	<b>27</b>
10.1 Definition . . . . .	27
10.2 Schedule and Execution . . . . .	27
10.2.1 Schedule . . . . .	28
10.2.2 Execution . . . . .	28
10.3 Deliverables . . . . .	29
10.4 Organization and Resources . . . . .	29
10.4.1 Example . . . . .	30
<b>A Verification Matrix</b>	<b>31</b>

# Data Management Test Plan

## 1 Introduction

In this document we lay out the verification and validation approach for LSST Data Management. In addition we outline some of the high level test milestones in Section 6 and our planned schedule for demonstrating interim verification status.

### 1.1 Objectives

We describe the test and verification approach for DM and describe various constraints and limitations in the testing to be performed. We also describe the validation tests to 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. Here we outline the required elements for those specifications as well as the tools we use to for continuous verification.

### 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 Science Validations in order to demonstrate the system's end-to-end capability against its design specifications. A large amount of informal science validation will be done in the the teams and documented in technical notes; in this test plan we are looking for validation of the broader system and specifically *operability* i.e. whether we can run this system every day for the 10 year planned survey with a reasonable level of operational support.

## 1.4 Applicable Documents

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

LPM-55 LSST Quality Assurance Plan  
LDM-294 DM Project Management Plan  
LDM-148 DM Architecture

## 1.5 References

- [1] **[LSE-29]**, Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2016, *LSST System Requirements*, LSE-29, URL <https://ls.st/LSE-29>
- [2] **[LSE-30]**, Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2016, *LSST System Requirements*, LSE-30, URL <https://ls.st/LSE-30>
- [3] **[LSE-81]**, Dubois-Felsmann, G., 2013, *LSST Science and Project Sizing Inputs*, LSE-81, URL <https://ls.st/LSE-81>
- [4] **[LSE-61]**, Dubois-Felsmann, G., 2016, *LSST Data Management Subsystem Requirements*, LSE-61, URL <https://ls.st/LSE-61>
- [5] **[LSE-82]**, Dubois-Felsmann, G., Lim, K.T., 2013, *Science and Project Sizing Inputs Explanation*, LSE-82, URL <https://ls.st/LSE-82>
- [6] **[LPM-17]**, Ivezić, Ž., The LSST Science Collaboration, 2011, *LSST Science Requirements Document*, LPM-17, URL <https://ls.st/LPM-17>
- [7] **[LSE-163]**, Jurić, M., et al., 2017, *LSST Data Products Definition Document*, LSE-163, URL <https://ls.st/LSE-163>
- [8] **[LDM-240]**, Kantor, J., Jurić, M., Lim, K.T., 2016, *Data Management Releases*, LDM-240, URL <https://ls.st/LDM-240>
- [9] **[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>

- [10] **[LDM-294]**, O'Mullane, W., Swinbank, J., Jurić, M., DMLT, 2017, *Data Management Organization and Management*, LDM-294, URL <https://ls.st/LDM-294>
- [11] **[LPM-55]**, Sweeney, D., McKercher, R., 2013, *Project Quality Assurance Plan*, LPM-55, URL <https://ls.st/LPM-55>
- [12] **[LSE-63]**, Tyson, T., DQA Team, Science Collaboration, 2017, *Data quality Assurance Plan: Requirements for the LSST Data Quality Assessment Framework*, LSE-63, URL <https://ls.st/LSE-63>

## 1.6 Definitions, Acronyms, and Abbreviations

Acronym	Description
DAX	Data Access Services
DBB	Data BackBone
DM	Data Management
DMCCB	DM Change Control Board
DRP	Data Release Production
EFD	Engineering Facilities Database
HSC	Hyper Suprime-Cam
ICD	Interface Control Document
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
NCSA	National Center for Supercomputing Applications
OCS	Observatory Control System
OPS	OPerationS
QA	Quality Assurance
QC	Quality Control
Qserv	Proprietary LSST Database system
SPR	Software Problem Report
SQuaRE	Science Quality and Reliability Engineering
SV	Science Validation
TBD	To Be Defined (Determined)
UX	User interface widget



VCD	Verification Control Document
-----	-------------------------------

## 2 Roles and Reporting

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

Testers report issues (SPRs) through the Data Management ticketing system (i.e. JIRA at the time of this document revision) and also write a test report (and/or provide any necessary configuration for automatic report generation).

The test reports will be used to populate the verification control document (see Section 3). We are monitoring the LSST Systems Engineer's approach to plan commissioning tests for LSST system-wide verification and will evaluate the merits of using the same toolchain for Data Management verification.

Operations rehearsals require an *ops rehearsal coordinator* to oversee the process. This is a distinct role from that of the tester. 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.

Tests and procedures will sometimes fail: a test specification may be re-run several times until it passes, but the report must include 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.

For large scale tests and rehearsals the DMCCB, or an individual designated by it, will be tasked to write up the findings as well as decide on timescales for re-running part or all of a test in case of failure or partial success.

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.

## 3 DM Verification Approach

Our approach towards verifying the Data Management requirements follows standard engineering practice. Each high level component will have at least one test specification defining a set of tests related to the design requirements for the component. These specifications are represented on the top of Figure 1. Any given requirement may have several tests associated with it in the specification; these tests may be phased to account for incremental delivery depending on the need for certain functionality at a specific time.

The test spec will cover all aspects of the test as outlined in Section 3.3. These high level test specifications may call out individual lower level test specifications where it makes sense (either technically or programmatically) to test lower-level components in isolation.

### 3.1 Reports

As we execute tests we will generate test reports on the pass/fail status of the individual tests related to specific requirements. This information will allow us to build a Verification Control Document (VCD) (shown at the right of Figure 1). The VCD will provide the fractional verification status of each DM requirement. These will also be rolled up to the (higher) level of OSS (Observatory System Specifications; LSE-30) requirements. Figure 1 currently calls for a report from each test spec. This report may be captured directly in e.g. JIRA: it does not necessarily correspond to a separate (e.g. Word or LaTeX) document.

In cases of reports that are generated via automatic (continuous) verification, the report may be in the format of a Jupyter Notebook that simultaneously can serve as test specification and test report and, in some cases, the test script itself. This is the preferred method, provided the notebook-as-report is satisfactorily captured in DocuShare.

### 3.2 Components Under Test

The components of the DM system are outlined in LDM-294 and detailed in LDM-148. At a high level these components are represented in Figure 2. 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.

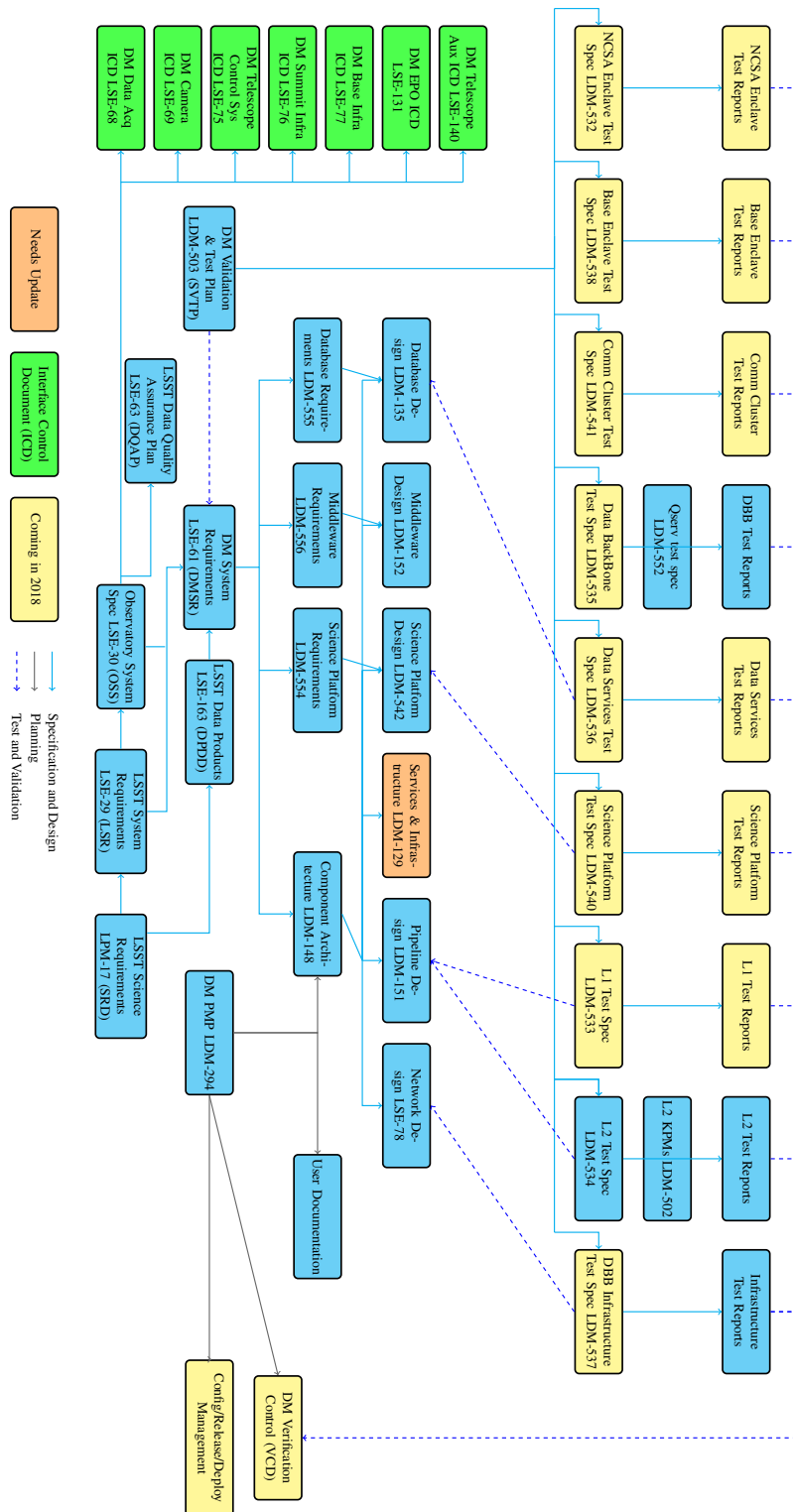


FIGURE 1: Documentation tree for DM software relating the high level documents to each other. (from LDM-294)

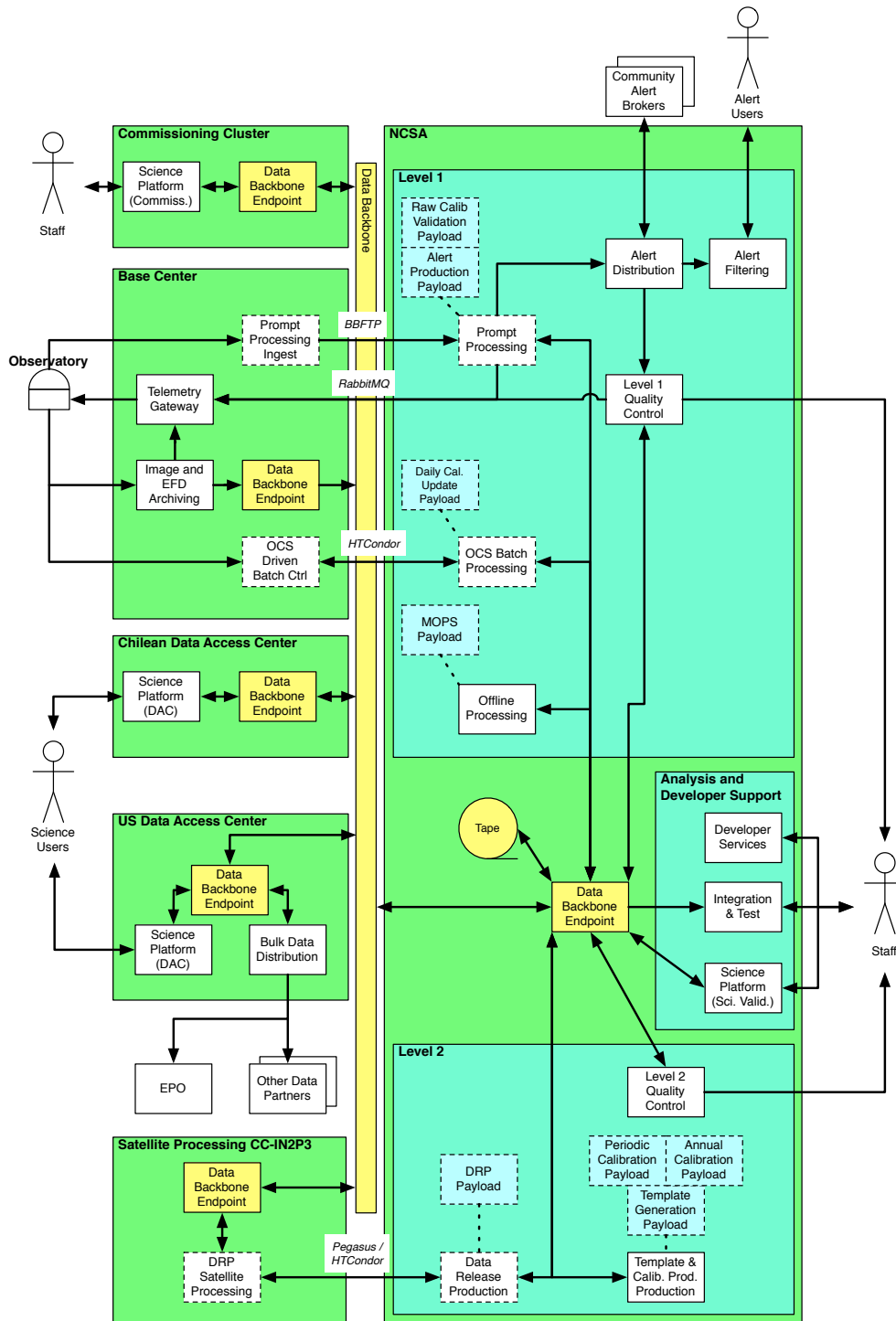


FIGURE 2: DM components as deployed during Operations. Where components are deployed in multiple locations, the connections between them are labeled with the relevant communication protocols. Science payloads are shown in blue. For details, refer to LDM-148.

TABLE 1: Components from LDM-148 with the test specifications to verify them.

<b>Component</b>	<b>Testing Spec</b>
<b>NCSA</b>	<b>LDM-532</b>
- L1 System	LDM-533
-- L1 Prompt Processing	TBD
-- L1 Alert Distribution	TBD
-- L1 Alert Filtering (mini Broker)	TBD
-- L1 Quality Control	TBD
-- L1 OCS Batch Processing	TBD
-- L1 Offline Processing	TBD
- L2 System	LDM-534
-- L2 QC	LDM-534
-- L2 Data Release	LDM-534
-- L2 Calibration Products	LDM-534
<b>Data Backbone</b>	<b>LDM-535</b>
- DBB Data Services	LDM-536
-- DBB Qserv	LDM-552
-- DBB Databases	TBD
- - DBB Image Database/Metadata Prov	TBD
-- DBB Data Butler Client	TBD
- DBB infrastructure	LDM-537
-- DBB Tape Archive	TBD
-- DBB Cache	TBD
-- DBB Data Endpoint	TBD
-- DBB Data Transport	TBD
-- Networks	TBD
<b>Base Center</b>	<b>LDM-538</b>
-- Prompt Processing Ingest	TBD
-- Telemetry Gateway	TBD
-- Image and EFD Archiving	TBD
-- OCS Driven Batch Control	TBD
<b>Data Access Center</b>	<b>LDM-539</b>
-- Bulk Data Distribution	TBD
-- Science Platform	LDM-540
-- Science Platform JupyterLab	TBD
-- Science Platform Portal	TBD
-- DAX VO+ Services	TBD
<b>Commissioning Cluster</b>	<b>LDM-541</b>
-- SuperTask	

The test items covered in this test plan are:

- Data Management 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.

### 3.3 Testing Specification Document Format

The testing specification documents will be drawn up in conjunction with the LSST Systems Engineer. In all cases they 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.
- A description of the environment in which the tests are 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.)

## 4 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. We preserve these distinctions where they have been made in, for example, the verification framework and automated metric harness. However for the purposes of pass/fail criteria, 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.

## 5 Constraints and Limitations

## 5.1 Procedural and Technical Limitations

- Verification is being done on the basis of precursor data sets such as HSC (see for example Section 7.2), and eventually with engineering data from the LSST camera test stands and commissioning camera. These are just a proxy for full-focal-plane on-site LSST data.
- 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.

## 5.2 Requirements Traceability Constraints

This section outlines the traceability of requirements through key LSST and Data Management documentation. In principle all DM requirements should be flowed down to LSE-61 (the DM System Requirements, or [DMSR](#)). In practice, while we are working to make that the reality, the current situation is outlined here.

### 5.2.1 Scientific

Some scientific requirements are captured in LSE-29 (the LSST System Requirements, or [LSR](#)) and flow down to LSE-30 (the Observatory System Specifications, or [OSS](#)). Work remains to flow them down from there to LSE-61.

Some requirements are also specified in LSE-163 (the Data Products Definition Document, or [DPDD](#)) and will flow down from there to LSE-61.

### 5.2.2 Computational

There are requirements in LSE-61 (aka [DMSR](#)) which captures the LSE-30 ([OSS](#)) requirements that DM is responsible for. These are:

- The primary computational performance flown down from LSE-29 ([LSR](#)) is OTT1 which



is the requirement to issue an alert within 60 seconds of exposure end.

DMS-REQ-0004  
LSR-REQ-0101

- Another requirement flown down from LSE-29 is calculation of orbits within 24 hours of the end of the observing night.
- There is a new (not yet baselined?) requirement for the calibration pipeline to reduce calibration observations within 1200 seconds.
- A nightly report on data quality, data management system performance and a calibration report have to be generated with 4 hours of the end of the night.

DMS-REQ-0004  
LSR-REQ-0104  
L1PublicT

calProcTime

DMS-REQ-0096  
dqReportComplTime

Work remains to flow down LSE-63, the Data Quality Assurance Plan, to LSE-61.

Note that there are no computational requirements on individual technical components such as data processing cluster availability, database data retrieval speeds, etc. There is, however, an upper limit on acceptable data loss, and there is a network availability requirement.

### 5.2.3 KPMs

As a proxy for validating the DM system, LDM-240, the—now obsolete—DM release plan, defined a set of Key Performance Metrics that the system could be verified against. KPMs were not formally flowed down from LSE-29 through LSE-30, although there is some overlap with LSE-29 requirements. In particular, the non-science KPMs only exist in LDM-240, although they are implicitly assumed in the sizing model presented in LSE-81 and LSE-82. Although other material in LDM-240 is now regarded as obsolete, these KPMs are still being tracked.

## 5.3 Interfaces

We will verify external interfaces to other subsystems and selected major internal interfaces. The ICDs describing external interfaces are curated in DocuShare Collection 5201.

## 6 Master Schedule

The schedule for testing the system until operations commence (currently 2022) is outlined in Table 3. These tests mark the major<sup>1</sup> milestones of the DM project. They are closely tied to major integration events for the overall LSST system, as shown in Figure 3.

Table 3: List of High Level integration tests for DM

ID	Date/Freq	Location	Title, Description
LDM-503-NLY	Nightly	Amazon	<b>Nightly Tests</b> Run all automated tests on all DM packages automatically.
LDM-503-WLY	Weekly	Amazon	<b>Integration tests</b> Basic Sanity check to make sure code compiles at no regressions have occurred and also pushing through a basic data set.
LDM-503-	TBD	NCSA	Interface tests The interface tests have to be planned and documented in a separate test plan that should include tests for each two parties on an interface (2by2 tests) as well as tests for all parties. Some of these will be covered again in E2E tests but before that we should be confident they work. <b>This includes internal and external interfaces.</b>
LDM-503-	TBD	NCSA + IN2P3	End to End Tests ?? Freeze software for Ops .. <a href="https://confluence.lsstcorp.org/display/DM/Data+Processing+End+to+End+Testing">https://confluence.lsstcorp.org/display/DM/Data+Processing+End+to+End+Testing</a> What is the status of these ?
LDM-503-1	2017-11-30	NCSA	<b>Science Platform with WISE data in PDAC</b> 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

<sup>1</sup>Level 2, in the parlance of LDM-294.

LDM-503-2	2017-11-30	NCSA	<b>HSC reprocessing</b> 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 and stopping production. Generate a validation data set for weekly integration and other tests.
LDM-503-3	2017-11-30	NCSA	<b>Alert generation validation</b> Validate the alert generation stack performance on several DECam and HSC datasets.
LDM-503-4	2018-02-01	NCSA	<b>Aux Tel DAQ integration functionality test</b> The production Aux Tel data acquisition hardware should be available in Tucson in 2018-02. We should prepare by testing the adjacent archive systems.
LDM-503-4b	2018-02-12	NCSA	<b>Test Report: Aux Tel DAQ interface Integration Verification and spectrograph operations rehearsal</b> The production Aux Tel data acquisition hardware should be available in Tucson in 2018-02. We should test integration with the adjacent archive systems.
LDM-503-5	2018-05-31	NCSA	<b>Alert distribution validation</b> Validate alert distribution system and mini-broker fed by live or simulated live data.
LDM-503-6	2018-06-30	NCSA	<b>DM ComCam interface verification readiness</b> ComCam will be in Tucson on 2018-07-24. The DM system must be ready to deal with it.
LDM-503-7	2018-08-31	NCSA	<b>Camera data processing</b> Partial camera data should be available to DM July 31st. We plan to test DM stack with it.
LDM-503-8	2018-11-30	NCSA	<b>Spectrograph data acquisition</b> Demonstrate that we can acquire (and process?) data from the spectrograph.
LDM-503-9	2018-11-30	NCSA	<b>Verification tests in advance of pre-ops rehearsal for commissioning #1</b> Test how the system will run during commissioning. Chuck requests that this initial test focus on ISR.
LDM-503-10	2019-02-28	NCSA	<b>DAQ validation</b> There is a project Milestone that DAQ/DM/Networks are available March 15th. We need to run tests in Feb to show this is ready.

LDM-503-11a	2019-10-20	NCSA	<b>DM ComCam operations readiness</b> ComCam will be in use in Nov . The DM system must be ready to deal with it.
LDM-503-11	2019-10-31	NCSA	<b>Verification tests in advance of pre-ops rehearsal for commissioning #2</b> 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? Exercise control loops.
LDM-503-12	2020-01-31	NCSA	<b>Verification tests in advance of pre-ops rehearsal for commissioning #3</b> Dress rehearsal: commissioning starts in April, so by this stage we should be ready to do everything needed.
LDM-503-13	2020-11-30	NCSA	<b>Verification test in advance of ops rehearsal DRP (ComCam data)</b> ComCam data will now be available. Demonstrate its use in producing a data release.
LDM-503-14	2021-03-31	NCSA	<b>DM Software for Science Verification</b> Science Verification starts in April. Demonstrate that all required DM software is available.
LDM-503-15	2021-11-29	NCSA	<b>Verification tests in advance of large scale ops rehearsal</b> Science Verification data will now be available. Demonstrate its use in producing a data release.
LDM-503-16	2022-02-28	NCSA	<b>Verification test in advance of full DRP ops rehearsal</b> Test readiness for operations.
LDM-503-17	2022-09-30	NCSA	<b>Verification tests of full DRP</b> Confirm readiness for operations.

## 7 Verification Tests

### 7.1 Science Platform with WISE data in PDAC (LDM-503-1)

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

From DAX: need to be clear about which WISE datasets are to be loaded – the data wrangling

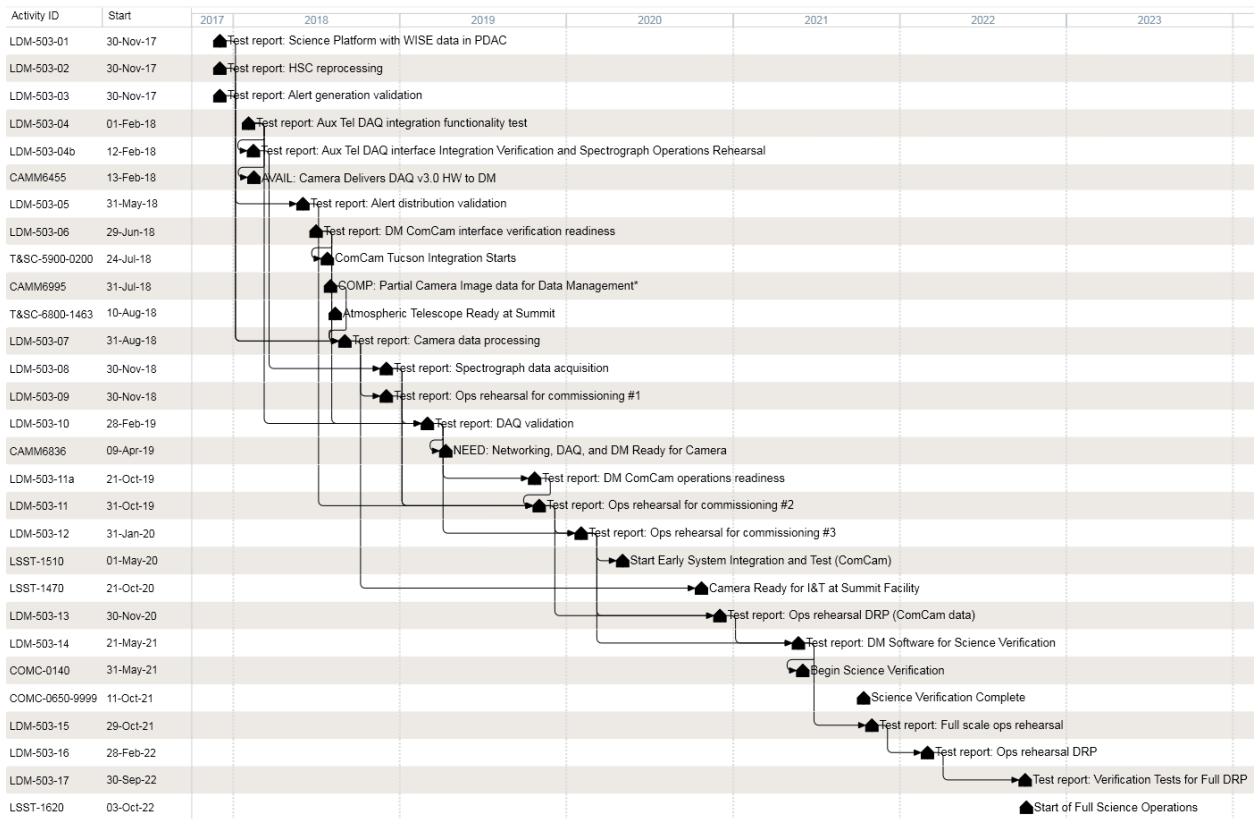


FIGURE 3: DM major milestones (LDM-503-x) in the LSST schedule.

effort required to download, inspect, convert, partition, and load each additional dataset is cumulatively non-trivial for DAX

## 7.2 HSC reprocessing (LDM-503-2)

### 7.2.1 Personnel

Jim Bosch, Robert Lupton, John Swinbank, Hsin-Fang Chiang.

### 7.2.2 Open issues

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

From the pipelines perspective, there's no new work involved here beyond the v13.0 release (at which point the HSC merge is complete and QA has been performed). Suggest we'd run this with the latest release as of the date of the test (so this is 14.N, where 14.0 is the end-of-S17 release). Again from pipelines, detailed definition of the "ops platform" is not necessary. Suggest that the plausible availability of services should drive the test plan in this case, rather than vice versa.

### 7.2.3 Datasets

During F17, we expect to continue testing and validation of Data Release Production algorithms primarily by repeated reprocessing of the first HSC Public Data Release (PDR1) on the LSST Verification Cluster (VC).

We expect to perform processing at three different scales:

- The full PDR1 dataset;

- A substantial fraction (nominally 10% of PDR1);
- The HSC “RC” dataset (a subset of PDR1<sup>2</sup> pre-selected for pipeline release testing).

The full PDR1 dataset consists of 6202 exposures, or 17 TB of raw data. It is now available in the /datasets/ filesystem on the VC (see RFC-297, DM-9683). One complete reprocessing of PDR1 requires around 200 TB of storage (see DM-8143); we therefore assume that 10% of PDR1 requires around 20 TB; we expect reprocessing the RC dataset to consume around 7 TB.

Again following DM-8143, we expect one complete reduction of PDR1 to consume around 750 core-weeks of CPU time (and, similarly, 75 core-weeks for a 10% fraction, or 25 core-weeks for the RC dataset). Note that:

- As of April 2017 there are 1152 cores in the VC, so we might reasonably expect that the entire data release can be processed in about 5 days.
- This assumes minimal inefficiency due to workflow; we expect wall-clock time to be rather higher.

**7.2.3.1 Automated Processing** We expect that some processing takes place automatically, without intervention or explicit request from the DRP team. In each case, processing makes use of the latest weekly release of the LSST stack, with the default configuration; in special circumstances, the DRP team may request an alternative version and/or configuration before the processing run starts.

The pipeline logic will be provided by the DRP team in whatever the currently-accepted standard for LSST data processing is. That is, we expect to use pipe\_drivers/ctrl\_pool style distribution middleware until the point at which a new solution, e.g. one based on SuperTask and Pegasus, becomes available. At that point, the DRP team is responsible for porting their pipelines to the new system.

We expect that regular execution of the relevant pipelines and checking for successful execution will take place outside the scope of DRP. We expect that failures at the execu-

<sup>2</sup>In fact, the existing RC dataset is not, in fact, all public. However, it should be straightforward to define a new RC-sized dataset which is.

tion middleware, hardware or networking layer will be resolved without the need for explicit pipelines intervention. We expect the DRP team to be responsible for triaging and resolving failures in pipeline logic, configuration, etc.

In the below, we suggest a calendar-based processing scheme. In practice, one which is tied to specific stack releases, rather than to the date, is likely preferable. However, implementing such a scheme would require rethinking the stack release procedure.

**7.2.3.1.1 PDR1** To be reprocessed every two months. The results of the last three jobs should be retained: in the steady state this will consume ~600 TB of storage.

**7.2.3.1.2 RC Dataset** To be reprocessed weekly. The results of the last four jobs should be retained: in the steady state this will consume ~28 TB of storage.

**7.2.3.2 Manual Processing** We request a mechanism by which developers may manually trigger processing jobs which will address broadly arbitrary subsets of HSC PDR1 with user specified software versions and configurations, e.g. as supplied through a configuration file (or shell script, etc).

Although DRP developers will be ultimately responsible for the successful execution of these jobs, we request support from NCSA in triaging failures which may be due to cluster or middleware issues.

**7.2.3.2.1 Storage** That the total storage requirement for such ad-hoc jobs during F17 will amount to no more than 200 TB. We suggest that this be provisioned in `/project/`, and that it follow the regular policies which apply to that filesystem.

**7.2.3.2.2 Compute** We expect to consume around 50 core-weeks per calendar week on ad hoc processing (that is, equivalent to two reductions of the RC dataset per week).

## 7.2.4 Calibration Products Production



**7.2.4.1 Datasets** We expect that data from both TS8 (RFC-301) and the 0.9 m at CTIO (RFC-313) continue to be regularly made available on the /datasets/ filesystem.

On the timescale of F17, we expect these datasets to total no more than 20 TB.

**7.2.4.2 Automated Processing** We do not request any automated processing of data for Calibration Products Producing during F17.

**7.2.4.3 Manual Processing** We expect that developers will manually trigger processing jobs which will address broadly arbitrary subsets of the TS8 & CTIO data with user specified software versions and configurations, e.g. as supplied through a configuration file (or shell script, etc).

Although DRP developers will be ultimately responsible for the successful execution of these jobs, we request support from NCSA in triaging failures which may be due to cluster or middleware issues.

**7.2.4.3.1 Storage** That the total storage requirement for such ad-hoc jobs during F17 will amount to no more than 50 TB. We suggest that this be provisioned in /project/, and that it follow the regular policies which apply to that filesystem.

**7.2.4.3.2 Compute** We expect to consume no more than 25 core-weeks per calendar week processing this data.

### 7.3 Alert generation validation (LDM-503-3)

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?

## 7.4 Aux Tel DAQ integration functionality test (LDM-503-4)

The production Aux Tel data acquisition hardware should be available in Tucson in 2018-02. We should prepare by testing the adjacent archive systems.

A minimal DM-only system that can archive mocked-up images and demonstrate that they can be retrieved, with provenance and metadata.

## 7.5 Test Report: Aux Tel DAQ interface Integration Verification and Spectrograph Operations Rehearsal (LDM-503-4b )

The production Aux Tel data acquisition hardware should be available in Tucson in 2018-02. We should test integration with the adjacent archive systems.

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

## 7.6 Alert distribution validation (LDM-503-5)

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

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

## 7.7 DM ComCam interface verification readiness (LDM-503-6)

ComCam will be in Tucson on 2018-07-24. The DM system must be ready to deal with it.

"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?

## 7.8 Camera data processing (LDM-503-7)

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

## 7.9 Spectrograph data acquisition (LDM-503-8)

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

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?

## 7.10 Verification tests in advance of pre-ops rehearsal for commissioning #1 (LDM-503-9)

Test how the system will run during commissioning. Chuck requests that this initial test focus on ISR.

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

## 7.11 DAQ validation (LDM-503-10)

There is a project Milestone that DAQ/DM/Networks are available March 15th. We need to run tests in Feb to show this is ready.

## 7.12 DM ComCam operations readiness (LDM-503-11a)

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

"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?

## 7.13 Verification tests in advance of pre-ops rehearsal for commissioning #2 (LDM-503-11)

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? Exercise control

loops.

### **7.14 Verification tests in advance of pre-ops rehearsal for commissioning #3 (LDM-503-12)**

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

### **7.15 Ops rehearsal DRP (ComCam data) (LDM-503-13)**

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

### **7.16 DM Software for Science Verification (LDM-503-14)**

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

### **7.17 Verification tests in advance of Full Operations rehearsal (LDM-503-15)**

Confirm readiness for operations.

### **7.18 Operations Rehearsal for Data Release Production (LDM-503-16)**

Test readiness for data release production.

## 7.19 Verification tests for full DRP (LDM-503-17)

Confirm readiness for data release production with fixes from commissioning.

## 8 Software Tools

A number of tools and development practices are in use in Data Management to ensure software quality and to verify requirements are met. These tools are used continuously (e.g. to measure key performance metrics routinely) or periodically (e.g. software release characterizations) and so will be well understood by the time the formal verification phase begins.

### 8.1 Continuous Integration and Unit Testing

Code is checked via a continuous integration (CI) service both for on-demand developer use and for verifying the quality of the master branch. Irrespective of supported platforms, we have a practice of verifying that the stack can run on at least two distinct operating systems/-platforms as portability is often a good indicator of maintainability. The CI service also permits verification that the codebase runs with different third party dependencies; for example we test that the python code runs both under (legacy) Python 2.7 and Python 3. This reduces the foreseeable technical debt of porting to Python 3 for operations.

Unit testing policy is described in the DM Developer guide under Unit Test Policy.

Roles and responsibilities in this area include:

- The DM Systems Engineering Team team is responsible for approving dependencies and setting strategy such as the approach to Python 3 portability.
- The DM Systems Engineering Team 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.

At the time of this revision we do not have unit test coverage tooling for Python. This will be remedied with an upcoming switch to the pytest framework.

## 8.2 Code Reviews

DM's process requires that every story resulting in code changes to the stack is reviewed prior to being merged to master. 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. DM's Code Review process is described in the DM Developer guide under the section DM Code Review and Merging Process.

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).

## 8.3 Automated Requirements Verification and KPM Measurement

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

- Calculating KPMs where available and alerting when they exceed specification.
- A 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 calculate the KPMs.
- SQuaRE is responsible for developing and operating the continuous metric verification services.
- Individual developers contribute non-KPM metrics as desired.

## 9 Operations Validation

Operations Validation of the system is done through Operations Rehearsals (and/or end-to-end tests). This may repeat some or all of a science validation exercise but in a more operational setting with a focus on operations. The proposed rehearsal dates are listed in Table 4.

Table 4: Operations rehearsals for Ops validation of DM

Date/Freq	Location	Title, Description
Oct 2018	NCSA	<b>Operations rehearsal for commissioning</b> With TBD weeks commissioning (lets say a week) – pick which parts of plan we could rehearse. Chuck suggests Instrument Signal Removal should be the focus of this (or the next rehearsal).
Oct 2019	NCSA	<b>Operations rehearsal #2 for commissioning</b> More complete rehearsal – where do the scientist look at quality data? How do they feed it back to the Telescope ? How do we create/update calibrations ? Exercises some of the control loops.
Jan 2020	Base	<b>Operations rehearsal #3 for commissioning</b> Dress rehearsal – Just like it will be April for the actual commissioning.
Dec 2020	NCSA	<b>Operations rehearsal data release processing (commissioning data)</b> Dress rehearsal – Just like it will be April for the actual commissioning.
Nov 2021	NCSA/Base	<b>Operations rehearsal</b> Rehearsals for real operations which start Nov 2021
Jan 2022	NCSA/Base	<b>Operations rehearsal data release processing</b> Full dress rehearsal for real data release

Aug 2022	NCSA	<b>Operations rehearsal for full data release processing (if needed).</b> Reconfirm operability of all updated software post commissioning.
----------	------	---

## 10 Science Validation

### 10.1 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.

### 10.2 Schedule and Execution



### 10.2.1 Schedule

DM SV 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. These follow a long-term roadmap of SV activities, linked to product delivery milestones in the DM's Construction Plan (see the table in Section 6). 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.

### 10.2.2 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.

### 10.3 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.

### 10.4 Organization and Resources

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.

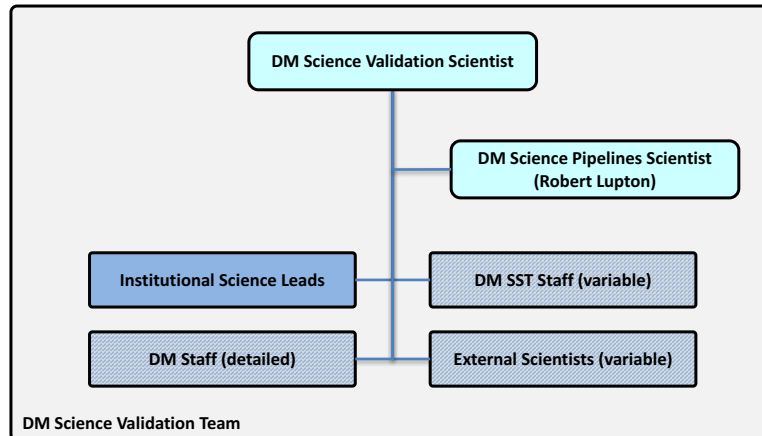


FIGURE 4: Organogram of the Data Management Science Validation Team. The group is chaired by the DM Science Validation Scientist, with the DM Science Pipelines Scientist and Institutional Science Leads making up the permanent membership. Depending on the SV activities being executed at any given time, 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.

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 be 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.

### 10.4.1 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.

## A Verification Matrix

The DM verification matrix may be found in LSE-61. A subset of the columns from the matrix are displayed here to indicate how we will verify DM requirements.

Requirement	Name	Method
DMS-REQ-0024	Raw Image Assembly	Demonstration
DMS-REQ-0018	Raw Science Image Data Acquisition	Test

DMS-REQ-0068	Raw Science Image Metadata	Test
DMS-REQ-0022	Crosstalk Corrected Science Image Data Acquisition	Test
DMS-REQ-0020	Wavefront Sensor Data Acquisition	Test
DMS-REQ-0265	Guider Calibration Data Acquisition	Demonstration
DMS-REQ-0004	Nightly Data Accessible Within 24 hrs	Test
DMS-REQ-0069	Processed Visit Images	Demonstration
DMS-REQ-0072	Processed Visit Image Content	Demonstration
DMS-REQ-0029	Generate Photometric Zeropoint for Visit Image	Demonstration
DMS-REQ-0030	Generate WCS for Visit Images	Test
DMS-REQ-0070	Generate PSF for Visit Images	Demonstration
DMS-REQ-0010	Difference Exposures	Demonstration
DMS-REQ-0074	Difference Exposure Attributes	Demonstration
DMS-REQ-0266	Exposure Catalog	Inspection
DMS-REQ-0269	DIASource Catalog	Demonstration
DMS-REQ-0270	Faint DIASource Measurements	
DMS-REQ-0271	DIAObject Catalog	Demonstration
DMS-REQ-0272	DIAObject Attributes	Demonstration
DMS-REQ-0273	SSObject Catalog	Demonstration
DMS-REQ-0317	DIAForcedSource Catalog	Demonstration
DMS-REQ-0274	Alert Content	Demonstration
DMS-REQ-0097	Level 1 Data Quality Report Definition	Demonstration
DMS-REQ-0099	Level 1 Performance Report Definition	Demonstration
DMS-REQ-0101	Level 1 Calibration Report Definition	Demonstration
DMS-REQ-0267	Source Catalog	Demonstration
DMS-REQ-0275	Object Catalog	Demonstration
DMS-REQ-0276	Object Characterization	Inspection
DMS-REQ-0046	Provide Photometric Redshifts of Galaxies	Inspection
DMS-REQ-0034	Associate Sources to Objects	Demonstration
DMS-REQ-0279	Deep Detection Coadds	Demonstration
DMS-REQ-0280	Template Coadds	Demonstration
DMS-REQ-0281	Multi-band Coadds	Demonstration
DMS-REQ-0278	Coadd Image Attributes	Demonstration
DMS-REQ-0047	Provide PSF for Coadded Images	Demonstration
DMS-REQ-0106	Coadded Image Provenance	Demonstration
DMS-REQ-0277	Coadd Source Catalog	Demonstration
DMS-REQ-0268	Forced-Source Catalog	Demonstration
DMS-REQ-0103	Produce Images for EPO	Demonstration
DMS-REQ-0130	Calibration Data Products	Demonstration
DMS-REQ-0132	Calibration Image Provenance	Demonstration

DMS-REQ-0059	Bad Pixel Map	Demonstration
DMS-REQ-0060	Bias Residual Image	Demonstration
DMS-REQ-0061	Crosstalk Correction Matrix	Demonstration
DMS-REQ-0282	Dark Current Correction Frame	Demonstration
DMS-REQ-0063	Monochromatic Flatfield Data Cube	Demonstration
DMS-REQ-0062	Illumination Correction Frame	Demonstration
DMS-REQ-0283	Fringe Correction Frame	Demonstration
DMS-REQ-0291	Query Repeatability	Demonstration
DMS-REQ-0292	Uniqueness of IDs Across Data Releases	Demonstration
DMS-REQ-0293	Selection of Datasets	Demonstration
DMS-REQ-0294	Processing of Datasets	Demonstration
DMS-REQ-0295	Transparent Data Access	Demonstration
DMS-REQ-0284	Level-1 Production Completeness	Demonstration
DMS-REQ-0131	Calibration Images Available 1 Hour Before Observing	Demonstration
DMS-REQ-0002	Transient Alert Distribution	Demonstration
DMS-REQ-0285	Level 1 Source Association	Demonstration
DMS-REQ-0286	SSObject Precovery	Demonstration
DMS-REQ-0287	DIASource Precovery	Demonstration
DMS-REQ-0288	Use of External Orbit Catalogs	Demonstration
DMS-REQ-0089	Solar System Objects Available within 24 hours	Demonstration
DMS-REQ-0096	Generate Data Quality Report within 4 hours	Demonstration
DMS-REQ-0098	Generate DMS Performance Report within 4 hours	Demonstration
DMS-REQ-0100	Generate Calibration Report within 4 hours	Demonstration
DMS-REQ-0289	Calibration Production Processing	Inspection
DMS-REQ-0006	Timely Publication of Level 2 Data Releases	Inspection
DMS-REQ-0290	Level 3 Data Import	Demonstration
DMS-REQ-0119	DAC resource allocation for Level 3 processing	Demonstration
DMS-REQ-0120	Level 3 Data Product Self Consistency	Inspection
DMS-REQ-0121	Provenance for Level 3 processing at DACs	Inspection
DMS-REQ-0125	Software framework for Level 3 catalog processing	Demonstration
DMS-REQ-0128	Software framework for Level 3 image processing	Demonstration
DMS-REQ-0308	Software Architecture to Enable Community Re-Use	Demonstration
DMS-REQ-0009	Simulated Data	Demonstration
DMS-REQ-0296	Pre-cursor, and Real Data	Demonstration
DMS-REQ-0032	Image Differencing	Demonstration
DMS-REQ-0033	Provide Source Detection Software	Demonstration
DMS-REQ-0043	Provide Calibrated Photometry	Demonstration
DMS-REQ-0042	Provide Astrometric Model	Demonstration
DMS-REQ-0052	Enable a Range of Shape Measurement Approaches	Demonstration

DMS-REQ-0160	Provide User Interface Services	Demonstration
DMS-REQ-0297	DMS Initialization Component	Demonstration
DMS-REQ-0155	Provide Data Access Services	
DMS-REQ-0298	Data Product Access	Demonstration
DMS-REQ-0299	Data Product Ingest	Demonstration
DMS-REQ-0300	Bulk Download Service	Demonstration
DMS-REQ-0065	Provide Image Access Services	Demonstration
DMS-REQ-0301	Control of Level-1 Production	Demonstration
DMS-REQ-0156	Provide Pipeline Execution Services	
DMS-REQ-0302	Production Orchestration	Demonstration
DMS-REQ-0303	Production Monitoring	Demonstration
DMS-REQ-0304	Production Fault Tolerance	Demonstration
DMS-REQ-0158	Provide Pipeline Construction Services	
DMS-REQ-0305	Task Specification	Inspection
DMS-REQ-0306	Task Configuration	Demonstration
DMS-REQ-0307	Unique Processing Coverage	Demonstration
DMS-REQ-0309	Raw Data Archiving Reliability	Demonstration
DMS-REQ-0094	Keep Historical Alert Archive	Demonstration
DMS-REQ-0310	Un-Archived Data Product Cache	Demonstration