LARGE SYNOPTIC SURVEY TELESCOPE

Large Synoptic Survey Telescope (LSST)

# LSST Data Management Science Acceptance Test Specification

L.P. Guy, W.M. Wood-Vasey, E. Bellm, J.F. Bosch,
G.P. Dubois-Felsmann, M.L. Graham, R. Gruendl, K.S. Krughoff,
K.-T. Lim, R.H. Lupton, C. Slater, G. Comoretto

LDM-639

Latest Revision: 2018-07-10

## Abstract

This document describes the detailed acceptance test specification for the LSST Data Management System.

# Change Record

| Version | Date | Description | Owner name |
|---|---|---|---|
| | 2018-06-18 | Initial docgen from Jira. Updated author list, added sections 1 and 2 | DocGen: G. Comoretto, Finalize: L. Guy |
| | 2018-06-22 | Improve test case specifications | DocGen: G. Comoretto, Finalize: L. Guy |

*Document curator:* Leanne Guy
*Document source location:* `https://github.com/lsst/ldm-639`
*Version from source repository:* 27f38f6

# Contents

# LSST Data Management Science Acceptance Test Specification

## 1 Introduction

This document is intended to specify the science acceptance test procedure for the LSST Data Management System.

This current version provides Test Cases for $\sim 35\%$ of the requirements. It does not yet provide full Test Plans for comprehensive testing nor identify the fraction of each requirement covered by the existing Test Cases. This documented will be updated as work continues on completing Test Cases, Test Plans, and requirements coverage.

### 1.1 Objectives

This document describes the test cases required to validate the Data Management System requirements described in the LSST DM Subsystem Requirements document LSE-61. A full description of this product is provided in the Data Management System Design document, LDM-148 with the science requirements detailed in the LSST Science Requirements Document LPM-17.

It identifies test cases and procedures for the tests as well as the pass/fail criteria for each test.

### 1.2 Scope

This document provides the science acceptance test plan for the whole Data Management System (DMS), as described by the Data Management System Requirements in LSE-61.

## 1.3 Applicable Documents

| | |
|---|---|
| LPM-17 | LSST Science Requirements Document |
| LDM-148 | LSST Data Management System Design |
| LDM-294 | LSST DM Organization & Management |
| LDM-503 | LSST DM Test Plan |
| LSE-61 | LSST DM Subsystem Requirements |
| LSE-163 | LSST Data Products Definition Document |
| LDM-151 | LSST DM Science Pipelines Design |
| LSE-180 | Level 2 Photometric Calibration for the LSST Survey |
| LSE-30 | LSST Observatory System Specifications |

## 1.4 References

[1] **[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

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

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

[4] **[LSE-180]**, Jones, L., 2013, *Level 2 Photometric Calibration for the LSST Survey*, LSE-180, URL https://ls.st/LSE-180

[5] **[LSE-163]**, Jurić, M., et al., 2017, *LSST Data Products Definition Document*, LSE-163, URL https://ls.st/LSE-163

[6] **[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

[7] **[LDM-294]**, O'Mullane, W., Swinbank, J., Jurić, M., DMLT, 2018, *Data Management Organization and Management*, LDM-294, URL https://ls.st/LDM-294

[8] **[LDM-503]**, O'Mullane, W., Swinbank, J., Jurić, M., Economou, F., 2018, *Data Management Test Plan*, LDM-503, URL https://ls.st/LDM-503

[9] **[LDM-151]**, Swinbank, J.D., et al., 2017, *Data Management Science Pipelines Design*, LDM-151, URL https://ls.st/LDM-151

# 2 Approach

This document describes the science acceptance tests for the integrated Data Management System, with a focus on whether the data products and functionality provided satisfy the requirements described in LSE-61.

The requirements from LSE-61 are extracted into the Jira "LSST Verification and Validation" Project managed through the Jira Test Management Plugin system. Each LSE-61 requirement leads to a Verification Element. Each Verification Element has one or more more Tests Cases. Each Test Case describes a Test Script to be executed, the coverage, pre-conditions, configuration, test results, and other details as specified by LDM-503. Test Scripts may have common set up and analysis steps. The Jira system allows for these steps to be shared shared by other Test Scripts. This improves the clarity and consistency across the Test Cases.

In this document, each Test Case is listed here with the Verification Element it tests, a summary of the Test Items exercised by the Test Case, and the detailed steps to be executed by the Test Case. Shared steps between Test Scripts have been explicitly written out to appear fully in each Test Case.

## 2.1 Features to be tested

All top-level requirements for the LSST Data Management System described in LSE-61 are to be tested, including

- Data Products
- Alert, Calibration and Data Release Production
- LSST science pipeline software and middleware
- LSST facilities including the data archive, base, summit, and the communications between them to accept science and engineering data

## 2.2 Features not to be tested

This document does not describe facilities for periodically generating or collecting key performance metrics (KPMs), except insofar as those KPMs are incidentally measured as part of executing the documented test cases.

## 2.3 Pass/fail criteria

The results of all tests will be assessed using the criteria described in LDM-503 §4.

Note that, when executing pipelines, tasks or individual algorithms, any unexplained or unexpected errors or warnings appearing in the associated log or on screen output must be described in the documentation for the system under test. Any warning or error for which this is not the case must be filed as a software problem report and filed with the DMCCB.

## 2.4 Suspension criteria and resumption requirements

Refer to individual test cases where applicable.

## 2.5 Naming convention

**LVV** : Is the label for the "LSST Verification and Validation" project in Jira.

**LVV-XXX** : Are Verification Elements, where XXX is the Verification Element identifier. Each Verification Element has at least one Test Case.

**LVV-TYYY** : Are Test Cases. Each Test Case is associated with a Verification Element, where YYY is the Test Case identifier.

The Verification Elements are drawn from LSE-61 requirements which have names of the form DMS-REQ-ZZZZ.

# 3 Test Cases Summary

| Jira Id | Test Name |
|---|---|
| LVV-T23 | Verify implementation of Test Storing Approximations of Per-pixel Metadata (DMS-REQ-0326) |
| LVV-T24 | Verify implementation of Computing Derived Quantities (DMS-REQ-0331) |
| LVV-T25 | Verify implementation of Denormalizing Database Tables (DMS-REQ-0332) |
| LVV-T26 | Verify implementation of Maximum Likelihood Values and Covariances (DMS-REQ-0333) |
| LVV-T27 | Verify implementation of Data Availability (DMS-REQ-0346) |
| LVV-T28 | Verify implementation of Measurements in catalogs (DMS-REQ-0347) |
| LVV-T29 | Verify implementation of Raw Science Image Data Acquisition (DMS-REQ-0018) |
| LVV-T30 | Verify implementation of Wavefront Sensor Data Acquisition (DMS-REQ-0020) |
| LVV-T31 | Verify implementation of Crosstalk Corrected Science Image Data Acquisition (DMS-REQ-0022) |
| LVV-T32 | Verify implementation of Raw Image Assembly (DMS-REQ-0024) |
| LVV-T33 | Verify implementation of Raw Science Image Metadata (DMS-REQ-0068) |
| LVV-T34 | Verify implementation of Guider Calibration Data Acquisition (DMS-REQ-0265) |
| LVV-T35 | Verify implementation of Nightly Data Accessible Within 24 hrs (DMS-REQ-0004) |
| LVV-T36 | Verify implementation of Difference Exposures (DMS-REQ-0010) |
| LVV-T37 | Verify implementation of Difference Exposure Attributes (DMS-REQ-0074) |
| LVV-T38 | Verify implementation of Processed Visit Images (DMS-REQ-0069) |
| LVV-T39 | Verify implementation of Generate Photometric Zeropoint for Visit Image (DMS-REQ-0029) |
| LVV-T40 | Verify implementation of Generate WCS for Visit Images (DMS-REQ-0030) |
| LVV-T41 | Verify implementation of Generate PSF for Visit Images (DMS-REQ-0070) |
| LVV-T42 | Verify implementation of Processed Visit Image Content (DMS-REQ-0072) |
| LVV-T43 | Verify implementation of Background Model Calculation (DMS-REQ-0327) |
| LVV-T44 | Verify implementation of Documenting Image Characterization (DMS-REQ-0328) |

| Jira Id | Test Name |
|---|---|
| LVV-T45 | Verify implementation of Level 1 Data Quality Report Definition (DMS-REQ-0097) |
| LVV-T46 | Verify implementation of Level 1 Performance Report Definition (DMS-REQ-0099) |
| LVV-T47 | Verify implementation of Level 1 Calibration Report Definition (DMS-REQ-0101) |
| LVV-T48 | Verify implementation of Exposure Catalog (DMS-REQ-0266) |
| LVV-T49 | Verify implementation of DIASource Catalog (DMS-REQ-0269) |
| LVV-T50 | Verify implementation of Faint DIASource Measurements (DMS-REQ-0270) |
| LVV-T51 | Verify implementation of DIAObject Catalog (DMS-REQ-0271) |
| LVV-T52 | Verify implementation of DIAObject Attributes (DMS-REQ-0272) |
| LVV-T53 | Verify implementation of SSObject Catalog (DMS-REQ-0273) |
| LVV-T54 | Verify implementation of Alert Content (DMS-REQ-0274) |
| LVV-T55 | Verify implementation of DIAForcedSource Catalog (DMS-REQ-0317) |
| LVV-T56 | Verify implementation of Characterizing Variability (DMS-REQ-0319) |
| LVV-T57 | Verify implementation of Calculating SSObject Parameters (DMS-REQ-0323) |
| LVV-T58 | Verify implementation of Matching DIASources to Objects (DMS-REQ-0324) |
| LVV-T59 | Verify implementation of Regenerating L1 Data Products During Data Release Processing (DMS-REQ-0325) |
| LVV-T60 | Verify implementation of Publishing predicted visit schedule (DMS-REQ-0353) |
| LVV-T61 | Verify implementation of Associate Sources to Objects (DMS-REQ-0034) |
| LVV-T62 | Verify implementation of Provide PSF for Coadded Images (DMS-REQ-0047) |
| LVV-T63 | Verify implementation of Produce Images for EPO (DMS-REQ-0103) |
| LVV-T64 | Verify implementation of Coadded Image Provenance (DMS-REQ-0106) |
| LVV-T65 | Verify implementation of Source Catalog (DMS-REQ-0267) |
| LVV-T66 | Verify implementation of Forced-Source Catalog (DMS-REQ-0268) |
| LVV-T67 | Verify implementation of Object Catalog (DMS-REQ-0275) |
| LVV-T68 | Verify implementation of Provide Photometric Redshifts of Galaxies (DMS-REQ-0046) |

| Jira Id | Test Name |
|---|---|
| LVV-T69 | Verify implementation of Object Characterization (DMS-REQ-0276) |
| LVV-T70 | Verify implementation of Coadd Source Catalog (DMS-REQ-0277) |
| LVV-T71 | Verify implementation of Detecting extended low surface brightness objects (DMS-REQ-0349) |
| LVV-T72 | Verify implementation of Coadd Image Method Constraints (DMS-REQ-0278) |
| LVV-T73 | Verify implementation of Deep Detection Coadds (DMS-REQ-0279) |
| LVV-T74 | Verify implementation of Template Coadds (DMS-REQ-0280) |
| LVV-T75 | Verify implementation of Multi-band Coadds (DMS-REQ-0281) |
| LVV-T76 | Verify implementation of All-Sky Visualization of Data Releases (DMS-REQ-0329) |
| LVV-T77 | Verify implementation of Best Seeing Coadds (DMS-REQ-0330) |
| LVV-T78 | Verify implementation of Persisting Data Products (DMS-REQ-0334) |
| LVV-T79 | Verify implementation of PSF-Matched Coadds (DMS-REQ-0335) |
| LVV-T80 | Verify implementation of Detecting faint variable objects (DMS-REQ-0337) |
| LVV-T81 | Verify implementation of Targeted Coadds (DMS-REQ-0338) |
| LVV-T82 | Verify implementation of Tracking Characterization Changes Between Data Releases (DMS-REQ-0339) |
| LVV-T83 | Verify implementation of Bad Pixel Map (DMS-REQ-0059) |
| LVV-T84 | Verify implementation of Bias Residual Image (DMS-REQ-0060) |
| LVV-T85 | Verify implementation of Crosstalk Correction Matrix (DMS-REQ-0061) |
| LVV-T86 | Verify implementation of Illumination Correction Frame (DMS-REQ-0062) |
| LVV-T87 | Verify implementation of Monochromatic Flatfield Data Cube (DMS-REQ-0063) |
| LVV-T88 | Verify implementation of Calibration Data Products (DMS-REQ-0130) |
| LVV-T89 | Verify implementation of Calibration Image Provenance (DMS-REQ-0132) |
| LVV-T90 | Verify implementation of Dark Current Correction Frame (DMS-REQ-0282) |
| LVV-T91 | Verify implementation of Fringe Correction Frame (DMS-REQ-0283) |
| LVV-T92 | Verify implementation of Processing of Data From Special Programs (DMS-REQ-0320) |
| LVV-T93 | Verify implementation of Level 1 Processing of Special Programs Data (DMS-REQ-0321) |
| LVV-T94 | Verify implementation of Special Programs Database (DMS-REQ-0322) |

| Jira Id | Test Name |
|---|---|
| LVV-T95 | Verify implementation of Constraints on Level Special Program Products Generation (DMS-REQ-0344) |
| LVV-T96 | Verify implementation of Query Repeatability (DMS-REQ-0291) |
| LVV-T97 | Verify implementation of Uniqueness of IDs Across Data Releases (DMS-REQ-0292) |
| LVV-T98 | Verify implementation of Selection of Datasets (DMS-REQ-0293) |
| LVV-T99 | Verify implementation of Processing of Datasets (DMS-REQ-0294) |
| LVV-T100 | Verify implementation of Transparent Data Access (DMS-REQ-0295) |
| LVV-T101 | Verify implementation of Transient Alert Distribution (DMS-REQ-0002) |
| LVV-T102 | Verify implementation of Solar System Objects Available Within Specified Time (DMS-REQ-0089) |
| LVV-T103 | Verify implementation of Generate Data Quality Report Within Specified Time (DMS-REQ-0096) |
| LVV-T104 | Verify implementation of Generate DMS Performance Report Within Specified Time (DMS-REQ-0098) |
| LVV-T105 | Verify implementation of Generate Calibration Report Within Specified Time (DMS-REQ-0100) |
| LVV-T106 | Verify implementation of Calibration Images Available Within Specified Time (DMS-REQ-0131) |
| LVV-T107 | Verify implementation of Level-1 Production Completeness (DMS-REQ-0284) |
| LVV-T108 | Verify implementation of Level 1 Source Association (DMS-REQ-0285) |
| LVV-T109 | Verify implementation of SSObject Precovery (DMS-REQ-0286) |
| LVV-T110 | Verify implementation of DIASource Precovery (DMS-REQ-0287) |
| LVV-T111 | Verify implementation of Use of External Orbit Catalogs (DMS-REQ-0288) |
| LVV-T112 | Verify implementation of Alert Filtering Service (DMS-REQ-0342) |
| LVV-T113 | Verify implementation of Performance Requirements for LSST Alert Filtering Service (DMS-REQ-0343) |
| LVV-T114 | Verify implementation of Pre-defined alert filters (DMS-REQ-0348) |
| LVV-T115 | Verify implementation of Calibration Production Processing (DMS-REQ-0289) |
| LVV-T116 | Verify implementation of Associating Objects across data releases (DMS-REQ-0350) |

| Jira Id | Test Name |
|---|---|
| LVV-T117 | Verify implementation of DAC resource allocation for Level 3 processing (DMS-REQ-0119) |
| LVV-T118 | Verify implementation of Level 3 Data Product Self Consistency (DMS-REQ-0120) |
| LVV-T119 | Verify implementation of Provenance for Level 3 processing at DACs (DMS-REQ-0121) |
| LVV-T120 | Verify implementation of Software framework for Level 3 catalog processing (DMS-REQ-0125) |
| LVV-T121 | Verify implementation of Software framework for Level 3 image processing (DMS-REQ-0128) |
| LVV-T122 | Verify implementation of Level 3 Data Import (DMS-REQ-0290) |
| LVV-T123 | Verify implementation of Access Controls of Level 3 Data Products (DMS-REQ-0340) |
| LVV-T124 | Verify implementation of Software Architecture to Enable Community Re-Use (DMS-REQ-0308) |
| LVV-T125 | Verify implementation of Simulated Data (DMS-REQ-0009) |
| LVV-T126 | Verify implementation Image Differencing (DMS-REQ-0032) |
| LVV-T127 | Verify implementation of Provide Source Detection Software (DMS-REQ-0033) |
| LVV-T128 | Verify implementation Provide Astrometric Model (DMS-REQ-0042) |
| LVV-T129 | Verify implementation of Provide Calibrated Photometry (DMS-REQ-0043) |
| LVV-T130 | Verify implementation of Enable a Range of Shape Measurement Approaches (DMS-REQ-0052) |
| LVV-T131 | Verify implementation of Provide User Interface Services (DMS-REQ-0160) |
| LVV-T132 | Verify implementation of Pre-cursor, and Real Data (DMS-REQ-0296) |
| LVV-T133 | Verify implementation of Provide Beam Projector Coordinate Calculation Software (DMS-REQ-0351) |
| LVV-T134 | Verify implementatino of Provide Image Access Services (DMS-REQ-0065) |
| LVV-T135 | Verify implementation of Provide Data Access Services (DMS-REQ-0155) |
| LVV-T136 | Verify implementation of Data Product and Raw Data Access (DMS-REQ-0298) |
| LVV-T137 | Verify implementation of Data Product Ingest (DMS-REQ-0299) |
| LVV-T138 | Verify implementation Bulk Download Service (DMS-REQ-0300) |

| Jira Id | Test Name |
|---|---|
| LVV-T139 | Verify implementation of Provide Pipeline Execution Services (DMS-REQ-0156) |
| LVV-T140 | Verify implementation of Production Orchestration (DMS-REQ-0302) |
| LVV-T141 | Verify implementation of Production Monitoring ( DMS-REQ-0303) |
| LVV-T142 | Verify implementation of Production Fault Tolerance (DMS-REQ-0304) |
| LVV-T143 | Verify implementation of Provide Pipeline Construction Services (DMS-REQ-0158) |
| LVV-T144 | Verify implementation of Task Specification (DMS-REQ-0305) |
| LVV-T145 | Verify implementation of Task Configuration (DMS-REQ-0306) |
| LVV-T146 | Verify implementation of DMS Initialization Component (DMS-REQ-0297) |
| LVV-T147 | Verify implementation pof Control of Level-1 Production (DMS-REQ-0301) |
| LVV-T148 | Verify implementation of Unique Processing Coverage (DMS-REQ-0307) |
| LVV-T149 | Verify implementation of Catalog Queries (DMS-REQ-0075) |
| LVV-T150 | Verify implementation of Maintain Archive Publicly Accessible (DMS-REQ-0077) |
| LVV-T151 | Verify implementation of Catalog Export Formats (DMS-REQ-0078) |
| LVV-T152 | Verify implementation of Keep Historical Alert Archive (DMS-REQ-0094) |
| LVV-T153 | Verify implementation of Provide Engineering and Facility Database Archive (DMS-REQ-0102) |
| LVV-T154 | Verify implementation of Raw Data Archiving Reliability (DMS-REQ-0309) |
| LVV-T155 | Verify implementation of Un-Archived Data Product Cache (DMS-REQ-0310) |
| LVV-T156 | Verify implementation of Regenerate Un-archived Data Products (DMS-REQ-0311) |
| LVV-T157 | Verify implementation Level 1 Data Product Access (DMS-REQ-0312) |
| LVV-T158 | Verify implementation Level 1 and 2 Catalog Access (DMS-REQ-0313) |
| LVV-T159 | Verify implementation of Regenerating Data Products from Previous Data Releases (DMS-REQ-0336) |
| LVV-T160 | Verify implementation of Providing a Precovery Service (DMS-REQ-0341) |
| LVV-T161 | Verify implementation of Logging of catalog queries (DMS-REQ-0345) |
| LVV-T162 | Verify implementation of Access to Previous Data Releases (DMS-REQ-0363) |
| LVV-T163 | Verify implementation of Data Access Services (DMS-REQ-0364) |
| LVV-T164 | Verify implementation of Operations Subsets (DMS-REQ-0365) |

| Jira Id | Test Name |
|---|---|
| LVV-T165 | Verify implementation of Subsets Support (DMS-REQ-0366) |
| LVV-T166 | Verify implementation of Access Services Performance (DMS-REQ-0367) |
| LVV-T167 | Verify implementation of Implementation Provisions (DMS-REQ-0368) |
| LVV-T168 | Verify implementation of Evolution (DMS-REQ-0369) |
| LVV-T169 | Verify implementation of Older Release Behavior (DMS-REQ-0370) |
| LVV-T170 | Verify implementation of Query Availability (DMS-REQ-0371) |
| LVV-T171 | Verify implementation of Pipeline Availability (DMS-REQ-0008) |
| LVV-T172 | Verify implementation of Optimization of Cost, Reliability and Availability (DMS-REQ-0161) |
| LVV-T173 | Verify implementation of Pipeline Throughput (DMS-REQ-0162) |
| LVV-T174 | Verify implementation of Re-processing Capacity (DMS-REQ-0163) |
| LVV-T175 | Verify implementation of Temporary Storage for Communications Links (DMS-REQ-0164) |
| LVV-T176 | Verify implementation of Infrastructure Sizing for ”catching up” (DMS-REQ-0165) |
| LVV-T177 | Verify implementation of Incorporate Fault-Tolerance (DMS-REQ-0166) |
| LVV-T178 | Verify implementation of Incorporate Autonomics (DMS-REQ-0167) |
| LVV-T179 | Verify implementation of Compute Platform Heterogeneity (DMS-REQ-0314) |
| LVV-T180 | Verify implementation of Data Management Unscheduled Downtime (DMS-REQ-0318) |
| LVV-T181 | Verify implementation of Summit Facility Data Communications (DMS-REQ-0168) |
| LVV-T182 | Verify implementation of Prefer Computing and Storage Down (DMS-REQ-0170) |
| LVV-T183 | Verify implementation DMS Communication with OCS (DMS-REQ-0315) |
| LVV-T184 | Verify implementation of Summit to Base Network (DMS-REQ-0171) |
| LVV-T185 | Verify implementation of Summit to Base Network Availability (DMS-REQ-0172) |
| LVV-T186 | Verify implementation of Summit to Base Network Reliability (DMS-REQ-0173) |
| LVV-T187 | Verify implementation of Summit to Base Network Secondary Link (DMS-REQ-0174) |

| Jira Id | Test Name |
|---|---|
| LVV-T188 | Verify implementation of Summit to Base Network Ownership and Operation (DMS-REQ-0175) |
| LVV-T189 | Verify implementation of Base Facility Infrastructure (DMS-REQ-0176) |
| LVV-T190 | Verify implementation of Base Facility Co-Location with Existing Facility (DMS-REQ-0178) |
| LVV-T191 | Verify implementation of Commissioning Cluster (DMS-REQ-0316) |
| LVV-T192 | Verify implementation of Base Wireless LAN (WiFi) (DMS-REQ-0352) |
| LVV-T193 | Verify implementation of Base to Archive Network (DMS-REQ-0180) |
| LVV-T194 | Verify implementation of Base to Archive Network Availability (DMS-REQ-0181) |
| LVV-T195 | Verify implementation of Base to Archive Network Reliability (DMS-REQ-0182) |
| LVV-T196 | Verify implementation of Base to Archive Network Secondary Link (DMS-REQ-0183) |
| LVV-T197 | Verify implementation of Archive Center (DMS-REQ-0185) |
| LVV-T198 | Verify implementation of Archive Center Disaster Recovery (DMS-REQ-0186) |
| LVV-T199 | Verify implementation of Archive Center Co-Location with Existing Facility (DMS-REQ-0187) |
| LVV-T200 | Verify implementation of Archive to Data Access Center Network (DMS-REQ-0188) |
| LVV-T201 | Verify implementation of Archive to Data Access Center Network Availability (DMS-REQ-0189) |
| LVV-T202 | Verify implementation of Archive to Data Access Center Network Reliability (DMS-REQ-0190) |
| LVV-T203 | Verify implementation of Archive to Data Access Center Network Secondary Link (DMS-REQ-0191) |
| LVV-T204 | Verify implementation of Access to catalogs for external Level 3 processing (DMS-REQ-0122) |
| LVV-T205 | Verify implementation of Access to input catalogs for DAC-based Level 3 processing (DMS-REQ-0123) |
| LVV-T206 | Verify implementation of Federation with external catalogs (DMS-REQ-0124) |

| Jira Id | Test Name |
|---|---|
| LVV-T207 | Verify implementation of Access to images for external Level 3 processing (DMS-REQ-0126) |
| LVV-T208 | Verify implementation of Access to input images for DAC-based Level 3 processing (DMS-REQ-0127) |
| LVV-T209 | Verify implementation of Data Access Centers (DMS-REQ-0193) |
| LVV-T210 | Verify implementation of Data Access Center Simultaneous Connections (DMS-REQ-0194) |
| LVV-T211 | Verify implementation of Data Access Center Geographical Distribution (DMS-REQ-0196) |
| LVV-T212 | Verify implementation of No Limit on Data Access Centers (DMS-REQ-0197) |

# 4 Test Cases

## 4.1 LVV-T23 - Verify implementation of Test Storing Approximations of Per-pixel Metadata (DMS-REQ-0326)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Simon Krughoff |

### 4.1.1 Test Items

Image depth and mask information shall be available in a parameterized approximate form in addition to a full per-pixel form.

### 4.1.2 Requirements

- LVV-157 - DMS-REQ-0326-V-01: Storing Approximations of Per-pixel Metadata

### 4.1.3 Precondition

Test data: A data repository containing a full DRP data reduction of the HCS PDR dataset.

### 4.1.4 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A "Data Butler" will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Create the coadd pixel level depth map for the HSC PDR dataset from step 1.

**Step 5**
Generate compressed representation of the pixel level depth map.

**Step 6**
Create the coadd pixel level mask map for the HSC PDR dataset from step 1.

**Step 7**
Generate compressed representation of the mask map.

**Step 8**
Sample randomly from both the pixel level and compressed depth maps. Compare the distribution of depths sampled from the pixel level depth map to that sampled from the compressed representation.

**Step 9**
Divide the mask planes into two groups: INFO and BAD. BAD flags are any that would cause a particular pixel to be excluded from processing: e.g. EDGE, SAT, BAD. Sample masks from both the pixel level mask map and the compressed mask map.

For each sample, compute sum(mask_pixel xor mask_compressed). Produce the distribution of the number of bits that differ between the samples.

Repeat for both the INFO flags and the BAD flags.

## 4.2 LVV-T24 - Verify implementation of Computing Derived Quantities (DMS-REQ-0331)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

### 4.2.1 Test Items

Common derived quantities shall be made available to end-users by either providing pre-computed columns or providing functions that can be used dynamically in queries. These should at least include the ability to calculate the reduced chi-squared of fitted models and make it as easy as possible to calculate color-color diagrams.

### 4.2.2 Requirements

- LVV-162 - DMS-REQ-0331-V-01: Computing Derived Quantities

### 4.2.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of

the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Load into DPDD+Science Platform

**Step 5**
Constructing color-color diagram and and fitting stellar locus in Science Platform.

**Step 6**
Invite three members of commissioning team to create color-color diagram from coadd catalogs based on merged coadd reference catalog.

## 4.3 LVV-T25 - Verify implementation of Denormalizing Database Tables (DMS-REQ-0332)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.3.1 Test Items

The database tables shall contain views presented to the users that will be appropriately denormalized for ease of use.

### 4.3.2 Requirements

- LVV-163 - DMS-REQ-0332-V-01: Denormalizing Database Tables

#### 4.3.3 Test Script

**Step 1**
Connect to the Science Platform's portal query interface.

**Step 2**
List the available views in the database.

**Step 3**
Take 20 sampled queries and determine which are easily done on views and which require complicated joins. Discuss the complicated ones and determine if any could be simplified by adding additional views.

### 4.4 LVV-T26 - Verify implementation of Maximum Likelihood Values and Covariances (DMS-REQ-0333)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.4.1 Test Items

Quantities delivered by all measurement algorithms shall include maximum likelihood values and covariances.

#### 4.4.2 Requirements

- LVV-164 - DMS-REQ-0333-V-01: Maximum Likelihood Values and Covariances

#### 4.4.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A "Data Butler" will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that maximum likelihood and covariant quantities are provide. Test and manually inspect that they are reasonable (finite, appropriately normed).

## 4.5 LVV-T27 - Verify implementation of Data Availability (DMS-REQ-0346)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.5.1 Test Item

All raw data used to generate any public data product (raw exposures, calibration frames, telemetry, configuration metadata, etc.) shall be kept and made available for download.

### 4.5.2 Requirements

- LVV-177 - DMS-REQ-0346-V-01: Data Availability

### 4.5.3 Test Script

**Step 1**
Invite two reviewers to review that plan that seems reasonable to expect the archiving and provision of raw data

**Step 2**
Pass a set of HSC data through (equal in size to the first public data release) the data backbone through ingest and provide interface

**Step 3**
Track the ingestion of AuxTel data during one month in 2018-2019 and verify delivery and test download.

## 4.6 LVV-T28 - Verify implementation of Measurements in catalogs (DMS-REQ-0347)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.6.1 Test Items

All catalogs shall record source measurements in flux units.

### 4.6.2 Requirements

- LVV-178 - DMS-REQ-0347-V-01: Measurements in catalogs

### 4.6.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘“bash

python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘’’

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 7**
A “Data Butler” will be initialized to access the repository.

**Step 8**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 9**

Verify that each of the single-visit, coadd, and difference image catalogs from HSC reporcessing and HiTS reprocessing (which may be the first source of regular difference images) provide measurements in flux units.

## 4.7 LVV-T29 - Verify implementation of Raw Science Image Data Acquisition (DMS-REQ-0018)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.7.1 Test Items

Delegate to Prompt Services (Ingest raw data from L1 Test Stand DAQ while simulating all modes)

### 4.7.2 Requirements

- LVV-8 - DMS-REQ-0018-V-01: Raw Science Image Data Acquisition

### 4.7.3 Test Script

**Step 1**

Ingest raw data from L1 Test Stand DAQ, simulating each observing mode

**Step 2**

Observe image metadata is present and queryable

## 4.8 LVV-T30 - Verify implementation of Wavefront Sensor Data Acquisition (DMS-REQ-0020)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.8.1 Test Items

Delegate to Prompt Services (Ingest wavefront sensor data from L1 Test Stand DAQ while simulating all modes)

### 4.8.2 Requirements

- LVV-9 - DMS-REQ-0020-V-01: Wavefront Sensor Data Acquisition

### 4.8.3 Test Script

**Step 1**
Ingest wavefront sensor data from L1 Test Stand DAQ while simulating all modes

**Step 2**
Observe wavefront sensor data and metadata archived

## 4.9 LVV-T31 - Verify implementation of Crosstalk Corrected Science Image Data Acquisition (DMS-REQ-0022)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.9.1 Test Items

Image and EFD Archiving, Prompt Processing & Delegate to Prompt Services (Ingest crosstalk corrected data from L1 Test Stand DAQ while simulating all modes)

### 4.9.2 Requirements

- LVV-10 - DMS-REQ-0022-V-01: Crosstalk Corrected Science Image Data Acquisition

### 4.9.3 Test Script

**Step 1**
Inject signals of different relative strength

**Step 2**
Apply Camera cross-talk correction

**Step 3**
Verify that DMS sytem can import the cross-talk corrected images

**Step 4**
Verify that images are corrected for crosstalk



## 4.10 LVV-T32 - Verify implementation of Raw Image Assembly (DMS-REQ-0024)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.10.1 Test Items

Delegate to Prompt Services (Ingest raw data from L1 Test Stand DAQ, observe image and metadata output)

### 4.10.2 Requirements

- LVV-11 - DMS-REQ-0024-V-01: Raw Image Assembly

### 4.10.3 Test Script

**Step 1**
Ingest data from L1 Camera Test Stand DAQ

**Step 2**
Simulate all different modes

**Step 3**
Verify that a raw image is constructed in correct format

**Step 4**
Verify that a raw image is constructed with correct metadata

## 4.11 LVV-T33 - Verify implementation of Raw Science Image Metadata (DMS-REQ-0068)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.11.1 Test Items

Ingest raw data from L1 Test Stand DAQ, observe image metadata is present and queryable

### 4.11.2 Requirements

- LVV-28 - DMS-REQ-0068-V-01: Raw Science Image Metadata

#### 4.11.3 Test Script

**Step 1**
Ingest raw data from L1 Test Stand DAQ, simulating each observing mode

**Step 2**
Observe image metadata is present and queryable

**Step 3**
Ingest data from L1 Camera Test Stand DAQ

**Step 4**
Simulate all different modes

**Step 5**
Verify that a raw image is constructed in correct format

**Step 6**
Verify that a raw image is constructed with correct metadata

**Step 7**
Verify that time of exposure start/end, site metadata, telescope metadata, and camera metadata are stored in DMS system.



### 4.12 LVV-T34 - Verify implementation of Guider Calibration Data Acquisition (DMS-REQ-0265)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

#### 4.12.1 Test Items

Ingest calibration frames from L1 Test Stand DAQ, execute CPP payloads, observe guider calibration products

#### 4.12.2 Requirements

- LVV-96 - DMS-REQ-0265-V-01: Guider Calibration Data Acquisition

#### 4.12.3 Test Script

**Step 1**
Ingest calibration frames from L1 Test Stand DAQ

**Step 2**
Execute CPP payloads

**Step 3**
Observe guider calibration products

### 4.13 LVV-T35 - Verify implementation of Nightly Data Accessible Within 24 hrs (DMS-REQ-0004)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

#### 4.13.1 Test Items

With the exception of alerts and Solar System Objects, all Level 1 Data Products shall be made public within time L1PublicT (LSR-REQ-0104) of the acquisition of the raw image data. Alerts shall be made available within time OTT1 (LSR-REQ-0101) from the conclusion of readout of the raw exposures used to generate each alert to the distribution of the alert to community distribution mechanisms. Solar System Object orbits shall, on average, be calculated before

the following night's observing finishes and the results shall be made available within time L1PublicT of those calculations being completed.

### 4.13.2 Requirements

- LVV-4 - DMS-REQ-0004-V-01: Nightly Data Accessible Within 24 hrs

### 4.13.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**

DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**

**These instructions need to be adjusted for Kubernetes**

Copy alert packets to local storage and change to that directory.

**Step 7**

**Step 8**

Start a consumer that monitors the full stream and logs only End of Partition status messages:

```
docker service create \
        --name monitor_full \
        --network alert_stream_default \
        --constraint node.role==worker \
        -e PYTHONUNBUFFERED=0 \
        alert_stream python bin/monitorStream.py full-stream > monitor_log.txt
```

**Step 9**

Start a consumer that monitors the full stream and logs a deserialized version of every Nth packet:

**adjust for printing Nth packets**

```
docker service create \
        --name monitor_full \
```

```
        --network alert_stream_default \
        --constraint node.role==worker \
        -e PYTHONUNBUFFERED=0 \
        alert_stream python bin/printStream.py full-stream > packet_log.txt
```

**Step 10**

Start a producer that reads alert packets from disk and loads them into the Kafka queue:

```
docker service create \
              --name sender \
              --network alert_stream_default \
              -v $PWD:/home/alert_stream/data:ro \
              -e PYTHONUNBUFFERED=0 \
              alert_stream python bin/sendAlertStream.py full-stream
```

**Step 11**

Examine output log files.

The monitor log should show end-of partition messages such as

```
topic:full-stream, partition:0, status:end, offset:1000, key:None, time:1528496269.734
```

And the packet log should show deserialized alert packets with contents matching the input packets.

**Step 12**

Time processing of data starting from (pre-ingested) raw files until an alert is available for distribution; verify that this time is less than OTT1.

**Step 13**
Time processing of data starting from (pre-ingested) raw files until the required data products are available in the Science Platform. Verify that this time is less than L1PublicT.

**Step 14**
Run MOPS on 1 night equivalent of LSST observing worth of precursor data and verify that Solar System Object orbits can be updated within 24 hours.

**Step 15**
Record time between completion of MOPS processing and availability of the updated SSObject catalogue through the Science Platform; verify this time is less than L1PublicT.

## 4.14 LVV-T36 - Verify implementation of Difference Exposures (DMS-REQ-0010)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.14.1 Test Items

The DMS shall create a Difference Exposure from each Processed Visit Image by subtracting a re-projected, scaled, PSF-matched Template Image in the same passband.

### 4.14.2 Requirements

- LVV-7 - DMS-REQ-0010-V-01: Difference Exposures

### 4.14.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
Demonstrate successful creation of a template image from HSC PDF and DECAM HiTS data. Demonstrate successful creation of a Difference Exposure for at least 10 other images from survey, ideally at a range of arimass. In particular, HiTS has 2013A u-band data. While the Blanco 4-m does have an ADC, there are still some chromatic effects and we should demonstrate that we can successfully produce Difference Exposures and templates for diferent airmass bins.

## 4.15 LVV-T37 - Verify implementation of Difference Exposure Attributes (DMS-REQ-0074)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.15.1 Test Items

For each Difference Exposure, the DMS shall store: the identify of the input exposures and related provenance information, and a set of metadata attributes including at least a representation of the PSF matching kernel used in the differencing

### 4.15.2 Requirements

- LVV-32 - DMS-REQ-0074-V-01: Difference Exposure Attributes

### 4.15.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl

‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
For each of HSC PDR and DECAM HiTS data: set up three different templates and run subtractions on 10 different images from at least two different filters. Verify that we can recover the provenance information about which template was used for each subtraction, which input images were used for that template, and that we can successfull extract the PSF matching kernel.



## 4.16 LVV-T38 - Verify implementation of Processed Visit Images (DMS-REQ-0069)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.16.1 Test Items

The DMS shall produce Processed Visit Images, in which the corresponding raw sensor array data has been trimmed of overscan and corrected for instrumental signature. Images

obtained in pairs during a standard visit are combined.

### 4.16.2 Requirements

- LVV-29 - DMS-REQ-0069-V-01: Processed Visit Images

### 4.16.3 Test Script

**Step 1**
Process HSC data, DECAM data. Verify that Processed Visit Images are generated at correct size and with significant instrumental artifacts removed.

**Step 2**
Run camera test stand data through full acquisition+backbone+ISR.

**Step 3**
Run simulated LSST data with calibrations through prompt processing system and inspect Processed Visit images to verify that they have been cleaned of significant artifacts and are of the correct, shape, and described orientation.

## 4.17 LVV-T39 - Verify implementation of Generate Photometric Zeropoint for Visit Image (DMS-REQ-0029)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.17.1 Test Items

Alert Production

### 4.17.2 Requirements

- LVV-12 - DMS-REQ-0029-V-01: Generate Photometric Zeropoint for Visit Image

### 4.17.3 Test Script

**Step 1**
Delegate to Alert Production

## 4.18 LVV-T40 - Verify implementation of Generate WCS for Visit Images (DMS-REQ-0030)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.18.1 Test Items

Alert Production

### 4.18.2 Requirements

- LVV-13 - DMS-REQ-0030-V-01: Generate WCS for Visit Images

### 4.18.3 Test Script

**Step 1**
Delegate to Alert Production

## 4.19 LVV-T41 - Verify implementation of Generate PSF for Visit Images (DMS-REQ-0070)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.19.1 Test Items

Alert Production

### 4.19.2 Requirements

- LVV-30 - DMS-REQ-0070-V-01: Generate PSF for Visit Images

### 4.19.3 Test Script

**Step 1**
Delegate to Alert Production

## 4.20 LVV-T42 - Verify implementation of Processed Visit Image Content (DMS-REQ-0072)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.20.1 Test Items

Alert Production

### 4.20.2 Requirements

- LVV-31 - DMS-REQ-0072-V-01: Processed Visit Image Content

### 4.20.3 Test Script

**Step 1**
Delegate to Alert Production

## 4.21 LVV-T43 - Verify implementation of Background Model Calculation (DMS-REQ-0327)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.21.1 Test Items

Alert Production

### 4.21.2 Requirements

- LVV-158 - DMS-REQ-0327-V-01: Background Model Calculation

### 4.21.3 Test Script

**Step 1**
Delegate to Alert Production

## 4.22 LVV-T44 - Verify implementation of Documenting Image Characterization (DMS-REQ-0328)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.22.1 Test Items

Alert Production

### 4.22.2 Requirements

- LVV-159 - DMS-REQ-0328-V-01: Documenting Image Characterization

#### 4.22.3 Test Script

**Step 1**
Delegate to Alert Production

### 4.23 LVV-T45 - Verify implementation of Level 1 Data Quality Report Definition (DMS-REQ-0097)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

#### 4.23.1 Test Items

QC System, Alert Production

#### 4.23.2 Requirements

- LVV-39 - DMS-REQ-0097-V-01: Level 1 Data Quality Report Definition

#### 4.23.3 Test Script

**Step 1**
Ingest raw data from L1 Test Stand DAQ, execute AP, load Prompt QC, observe telemetry and report

### 4.24 LVV-T46 - Verify implementation of Level 1 Performance Report Definition (DMS-REQ-0099)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.24.1 Test Items

Prompt Processing, QC System

### 4.24.2 Requirements

- LVV-41 - DMS-REQ-0099-V-01: Level 1 Performance Report Definition

### 4.24.3 Test Script

**Step 1**
Execute single-day operations rehearsal, observe report

## 4.25 LVV-T47 - Verify implementation of Level 1 Calibration Report Definition (DMS-REQ-0101)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.25.1 Test Items

OCS Driven Batch, Raw Calibration Validation, Daily Calibration Products Update

### 4.25.2 Requirements

- LVV-43 - DMS-REQ-0101-V-01: Level 1 Calibration Report Definition

### 4.25.3 Test Script

**Step 1**
Execute single-day operations rehearsal, observe report

## 4.26 LVV-T48 - Verify implementation of Exposure Catalog (DMS-REQ-0266)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.26.1 Test Items

Precursor data, execute AP, load results, observe catalog contents

### 4.26.2 Requirements

- LVV-97 - DMS-REQ-0266-V-01: Exposure Catalog

### 4.26.3 Test Script

**Step 1**
Verify that Exposure Catalogs contained required elements

**Step 2**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

## 4.27 LVV-T49 - Verify implementation of DIASource Catalog (DMS-REQ-0269)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.27.1 Test Items

Precursor data, execute AP, load results, observe catalog contents

### 4.27.2 Requirements

- LVV-100 - DMS-REQ-0269-V-01: DIASource Catalog

### 4.27.3 Test Script

**Step 1**
Verify that products are produced for DIASource catalog

**Step 2**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 3**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 4**

A "Data Butler" will be initialized to access the repository.

**Step 5**

For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 6**

DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

## 4.28 LVV-T50 - Verify implementation of Faint DIASource Measurements (DMS-REQ-0270)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.28.1 Test Items

Delegate to Alert Production (Precursor data, execute AP, observe measurements are present)

### 4.28.2 Requirements

- LVV-101 - DMS-REQ-0270-V-01: Faint DIASource Measurements

### 4.28.3 Precondition

Input Data
DECam HiTS data.

### 4.28.4 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
As an example of selecting with constrains, Re-run source detection as an afterburner to se-

lect isolated sources (defined as more than 2 arcseconds away from any other objects in the single-image-depth catalog) that are fainter than the fiducial transSNR cut.

## 4.29 LVV-T51 - Verify implementation of DIAObject Catalog (DMS-REQ-0271)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.29.1 Test Items

Precursor data, execute AP, load results, observe catalog contents

### 4.29.2 Requirements

- LVV-102 - DMS-REQ-0271-V-01: DIAObject Catalog

### 4.29.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T17 - AG-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
DIASource records will be accessed by querying the Butler, then examined interactively at a Python prompt.

**Step 4**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 5**

The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 6**
A “Data Butler” will be initialized to access the repository.

**Step 7**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 8**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 9**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T17 - AG-00-00).

**Step 10**
sqlite3 or Python’s sqlalchemy module will be used to access the Level 1 database.

**Step 11**
Verify that DIAObjects have diaNearbyObjMaxStar and diaNearbyObjMaxGalaxies that point to the Object catalog and are within dianNearbyObjRadius; the probability of association; and

the required DIAObject properties.

## 4.30 LVV-T52 - Verify implementation of DIAObject Attributes (DMS-REQ-0272)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.30.1 Test Items

Alert Production & Delegate to Alert Production

### 4.30.2 Requirements

- LVV-103 - DMS-REQ-0272-V-01: DIAObject Attributes

### 4.30.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A "Data Butler" will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.



## 4.31 LVV-T53 - Verify implementation of SSObject Catalog (DMS-REQ-0273)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.31.1 Test Items

Execute AP on precursor data, execute MOPS, load results, observe catalog contents

### 4.31.2 Requirements

- LVV-104 - DMS-REQ-0273-V-01: SSObject Catalog

### 4.31.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-

00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
Run the MOPS pipeline on the Prompt Products database.

**Step 7**
Inspect SSObject catalog and verify the presence of the required elements (LVV-104).

## 4.32 LVV-T54 - Verify implementation of Alert Content (DMS-REQ-0274)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.32.1 Test Items

Delegate to Alert Production

### 4.32.2 Requirements

- LVV-105 - DMS-REQ-0274-V-01: Alert Content

### 4.32.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A "Data Butler" will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
**These instructions need to be adjusted for Kubernetes**

Copy alert packets to local storage and change to that directory.

**Step 7**

**Step 8**
Start a consumer that monitors the full stream and logs only End of Partition status messages:

```
docker service create \
        --name monitor_full \
        --network alert_stream_default \
        --constraint node.role==worker \
        -e PYTHONUNBUFFERED=0 \
        alert_stream python bin/monitorStream.py full-stream > monitor_log.txt
```

**Step 9**
Start a consumer that monitors the full stream and logs a deserialized version of every Nth

packet:

**adjust for printing Nth packets**

```
docker service create \
        --name monitor_full \
        --network alert_stream_default \
        --constraint node.role==worker \
        -e PYTHONUNBUFFERED=0 \
        alert_stream python bin/printStream.py full-stream > packet_log.txt
```

**Step 10**

Start a producer that reads alert packets from disk and loads them into the Kafka queue:

```
docker service create \
              --name sender \
              --network alert_stream_default \
              -v $PWD:/home/alert_stream/data:ro \
              -e PYTHONUNBUFFERED=0 \
              alert_stream python bin/sendAlertStream.py full-stream
```

**Step 11**

Examine output log files.

The monitor log should show end-of partition messages such as

```
topic:full-stream, partition:0, status:end, offset:1000, key:None, time:1528496269.734
```

And the packet log should show deserialized alert packets with contents matching the input packets.

**Step 12**
Examine the serialized alert packets to confirm the presence of the required elements (LVV-105).

## 4.33 LVV-T55 - Verify implementation of DIAForcedSource Catalog (DMS-REQ-0317)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.33.1 Test Items

Precursor data, execute AP, load results, observe catalog contents

### 4.33.2 Requirements

- LVV-148 - DMS-REQ-0317-V-01: DIAForcedSource Catalog

### 4.33.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘“bash

```
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘
```

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.



## 4.34 LVV-T56 - Verify implementation of Characterizing Variability (DMS-REQ-0319)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.34.1 Test Items

Delegate to Alert Production

### 4.34.2 Requirements

- LVV-150 - DMS-REQ-0319-V-01: Characterizing Variability

### 4.34.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

```bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
```

and any errors or failures reported.

**Step 3**
A "Data Butler" will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing

it using sqlite3 and executing appropriate SQL queries.

**Step 6**
Verify that the issued alerts contain measurements during the diaCharacterizationCutoff.

## 4.35 LVV-T57 - Verify implementation of Calculating SSObject Parameters (DMS-REQ-0323)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.35.1 Test Items

Precursor data, execute MOPS, load results, observe functions usable in queries

### 4.35.2 Requirements

- LVV-154 - DMS-REQ-0323-V-01: Calculating SSObject Parameters

### 4.35.3 Test Script

**Step 1**
Run the MOPS pipeline on the Prompt Products database.

**Step 2**

**Step 3**
Inspect SSObject catalog and verify the presence of the required elements (LVV-104).

**Step 4**
Computer the phase angle, reduced and absolute asteroid magnitudes for objects identified in SSObject Catalog

## 4.36 LVV-T58 - Verify implementation of Matching DIASources to Objects (DMS-REQ-0324)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.36.1 Test Items

Execute DRP and AP on precursor data, load results, confirm crossmatch table or view is present

### 4.36.2 Requirements

- LVV-155 - DMS-REQ-0324-V-01: Matching DIASources to Objects

### 4.36.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-

00-00).

**Step 5**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 6**
A “Data Butler” will be initialized to access the repository.

**Step 7**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 8**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 9**
Verify that a cross-match table between the Prompt DIASources and DRP Objects is available.

## 4.37 LVV-T59 - Verify implementation of Regenerating L1 Data Products During Data Release Processing (DMS-REQ-0325)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.37.1 Test Items

Data Release Production

### 4.37.2 Requirements

- LVV-156 - DMS-REQ-0325-V-01: Regenerating L1 Data Products During Data Release Processing

### 4.37.3 Test Script

**Step 1**
Execute DRP

**Step 2**
Observe production of difference image data products

## 4.38 LVV-T60 - Verify implementation of Publishing predicted visit schedule (DMS-REQ-0353)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.38.1 Test Items

Delegate to PPP (Pointing Prediction Publishing)

### 4.38.2 Requirements

- LVV-184 - DMS-REQ-0353-V-01: Publishing predicted visit schedule

### 4.38.3 Test Script

**Step 1**

## 4.39 LVV-T61 - Verify implementation of Associate Sources to Objects (DMS-REQ-0034)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.39.1 Test Items

Data Release Production

### 4.39.2 Requirements

- LVV-16 - DMS-REQ-0034-V-01: Associate Sources to Objects

### 4.39.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of

the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that sources have objects

**Step 5**
Verify that objects list sources that seem reasonably near them.

## 4.40 LVV-T62 - Verify implementation of Provide PSF for Coadded Images (DMS-REQ-0047)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.40.1 Test Items

Data Release Production

### 4.40.2 Requirements

- LVV-20 - DMS-REQ-0047-V-01: Provide PSF for Coadded Images

### 4.40.3 Precondition

Fully covered by preconditions for LVV-T16.

### 4.40.4 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00)

**Step 2**
A "Data Butler" will be initialized to access the repository.

**Step 3**
For each combination of tract/patch/filter, the PVI will be retrieved from the Butler, and the existence of all components described in Test items section §4.6.2 will be verified.

**Step 4**
Scripts from the pipe_analysis package will be run on every visit to check for the presence of data products and make plots

**Step 5**
Ten patches will be chosen at random and inspected by eye for unmasked artifacts.

**Step 6**
Select Objects classified as point sources on 10 different coadd images (including all bands). Evaluate the PSF model at the positions of these Objects, and verify that subtracting a scaled version of the PSF model from the coadd image yields residuals consistent with pure noise.

## 4.41 LVV-T63 - Verify implementation of Produce Images for EPO (DMS-REQ-0103)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Gregory Dubois-Felsmann |

### 4.41.1 Test Items

Data Release Production

### 4.41.2 Requirements

- LVV-45 - DMS-REQ-0103-V-01: Produce Images for EPO

### 4.41.3 Precondition

This test will verify that the image data products called out in LSE-131 have been produced. In order for that to be successful, as a precondition the inputs to that production must exist. As the only currently mandated image data production in LSE-131 is that of a color all-sky HiPS map down to 1 arcsecond resolution, the prerequisite inputs to that must be the single-filter coadds in the bands required by the yet-to-be-specified color prescription.

Depending on the test dataset used for different runs of this test over time, e.g., precursor or LSST-commissioning data, the size of the resulting HiPS image map will vary.

### 4.41.4 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that a HiPS image map covering the LSST survey area, with a limiting depth yielding 1 arcsecond resolution, has been produced matching the color prescriptions provided by EPO (in updates to LSE-131 which are expected to be made ”once ComCam data is available”).

**Step 5**
Place the image map in a location accessible to a Firefly and an Aladin Lite client, ideally with the client running in the EPO data systems environment.

**Step 6**

Use Firefly to manually explore the image map at the largest scales to verify coverage of the entire sky. Sample in various locations to confirm the 1 arcsecond maximum depth.
Confirm using Aladin Lite that the format of the image map is supported by this common community tool.

**Step 7**
Verify programmatically, perhaps both by sampling a variety of locations, and by counting the tiles created at the 1-arcsecond-resolution depth, that the map is complete and meets its specifications.

**Step 8**
Apply an IVOA-community HiPS service validation tool, if available, to the service location.

**Step 9**
Verify that the HiPS map created is in a location accessible to the EPO data systems.



## 4.42 LVV-T64 - Verify implementation of Coadded Image Provenance (DMS-REQ-0106)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.42.1 Test Items

Data Release Production

### 4.42.2 Requirements

- LVV-46 - DMS-REQ-0106-V-01: Coadded Image Provenance

### 4.42.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A "Data Butler" will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Query and verify provenance of input images, and software versions that went into producing stack.

**Step 5**
Test re-generating 10 different coadds tract+patches based on the provenance image given

## 4.43 LVV-T65 - Verify implementation of Source Catalog (DMS-REQ-0267)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.43.1 Test Items

Data Release Production, Parallel Distributed Database

### 4.43.2 Requirements

- LVV-98 - DMS-REQ-0267-V-01: Source Catalog

### 4.43.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.



## 4.44 LVV-T66 - Verify implementation of Forced-Source Catalog (DMS-REQ-0268)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.44.1 Test Items

Precursor data, execute DRP, load results, observe catalog contents

### 4.44.2 Requirements

- LVV-99 - DMS-REQ-0268-V-01: Forced-Source Catalog

### 4.44.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 7**
A “Data Butler” will be initialized to access the repository.

**Step 8**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 9**
Verify that there exist entries in the forced-photometry table for all coadd objects for the PVIs on which the object should appear.

**Step 10**
Verify that there exist entries in a forced-photometry table for each image for all DIAObjects.



## 4.45 LVV-T67 - Verify implementation of Object Catalog (DMS-REQ-0275)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.45.1 Test Items

Precursor data, execute DRP, load results, observe catalog contents

### 4.45.2 Requirements

- LVV-106 - DMS-REQ-0275-V-01: Object Catalog

### 4.45.3 Precondition

Input Data

DECam HiTS data (raw science images and master calibrations)
HSC "RC2" data (raw science images and master calibrations)

### 4.45.4 Test Script

**Step 1**
load LSST DM Stack

**Step 2**
Run the single-frame processing and self-calibration steps of the DRP pipeline.

**Step 3**
Insert simulated sources into all single-frame images, including:

- static objects (e.g. galaxies), including some too faint to be detectable in single-epoch images;
- objects with static positions that are sufficiently bright and variable that they should be detectable in single-epoch difference images;
- transient objects that appear in only a few epochs;
- stars with significant proper motions and parallaxes, some below the single-epoch detection limit
- simulated solar system objects with orbits that can be constrained from just the epochs in the test dataset

**Step 4**
Run all remaining DRP pipeline steps.

**Step 5**
Load data into DRP database

**Step 6**
Verify that the injected simulated objects are recovered at a rate consistent with their S/N

*when not blended with each other or real objects*, and that flags indicating how each Object was detected are consistent with their properties:

- static objects should be detected in coadds only (not difference images)
- static-position/variable-flux objects should be detected in coadds and possibly difference images
- transient objects should be detected in difference images only
- stars with significant proper motions may be detected in either coadds or difference images
- solar system objects should be detected in difference images only.

## 4.46 LVV-T68 - Verify implementation of Provide Photometric Redshifts of Galaxies (DMS-REQ-0046)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.46.1 Test Items

Precursor data, execute DRP, load results, observe catalog contents

### 4.46.2 Requirements

- LVV-19 - DMS-REQ-0046-V-01: Provide Photometric Redshifts of Galaxies

### 4.46.3 Precondition

Input Data
HSC Public Data Release (raw science images, master calibrations)
Assorted public spectroscopic catalogs and high-accuracy photometric redshift catalogs in the HSC PDR footprint.

### 4.46.4 Test Script

**Step 1**
Run DRP processing steps through (at least) final galaxy photometry measurements.

**Step 2**
Train photometric redshift algorithm(s) on spectroscopic and high-accuracy photometric redshift catalogs.

**Step 3**
Estimate photometric redshifts for all Objects generated by DRP processing.

**Step 4**
Load into DRP Database

**Step 5**
Inspect database to verify that photometric redshifts are present for all objects

## 4.47 LVV-T69 - Verify implementation of Object Characterization (DMS-REQ-0276)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.47.1 Test Items

Data Release Production, Parallel Distributed Database

### 4.47.2 Requirements

- LVV-107 - DMS-REQ-0276-V-01: Object Characterization

### 4.47.3 Test Script

**Step 1**
Precursor data, execute DRP, load results, observe catalog contents

## 4.48 LVV-T70 - Verify implementation of Coadd Source Catalog (DMS-REQ-0277)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.48.1 Test Items

Precursor data, execute DRP, load results, observe catalog contents

### 4.48.2 Requirements

- LVV-108 - DMS-REQ-0277-V-01: Coadd Source Catalog

### 4.48.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that there exists a catalog of merged sources.

## 4.49 LVV-T71 - Verify implementation of Detecting extended low surface brightness objects (DMS-REQ-0349)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.49.1 Test Items

Data Release Production

### 4.49.2 Requirements

- LVV-180 - DMS-REQ-0349-V-01: Detecting extended low surface brightness objects

### 4.49.3 Precondition

Input Data
HSC ”RC2” data (raw science images and master calibrations)

### 4.49.4 Test Script

**Step 1**
load LSST DM Stack

**Step 2**
Run the single-frame processing and self-calibration steps of the DRP pipeline.

**Step 3**
Insert simulated low-surface-brightness galaxies (with exponential profiles) consistently into all calibrated single-epoch images.

**Step 4**
Run all remaining DRP pipeline steps.

**Step 5**
Load data into DRP database

**Step 6**
Verify that the injected simulated objects are recovered at a rate consistent with their S/N and true profile *when not blended with each other or real objects.*

## 4.50 LVV-T72 - Verify implementation of Coadd Image Method Constraints (DMS-REQ-0278)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.50.1 Test Items

Verify the implementation of how Coadd images are created.

### 4.50.2 Requirements

- LVV-109 - DMS-REQ-0278-V-01: Coadd Image Method Constraints

### 4.50.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that coadds were created following specification

## 4.51 LVV-T73 - Verify implementation of Deep Detection Coadds (DMS-REQ-0279)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.51.1 Test Items

Data Release Production

### 4.51.2 Requirements

- LVV-110 - DMS-REQ-0279-V-01: Deep Detection Coadds

### 4.51.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of

the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify through inspection that per-filter coadds exist for each tract+patch possible

**Step 5**
Verify through inspection that the images used to generate those coadds met specified conditions

**Step 6**
Visually inspect a subset of the coadds to verify that they visually appear reasonable and to be from good quality data.

## 4.52 LVV-T74 - Verify implementation of Template Coadds (DMS-REQ-0280)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.52.1 Test Items

Precursor or simulate data, execute Template Generation, observe image products

### 4.52.2 Requirements

- LVV-111 - DMS-REQ-0280-V-01: Template Coadds

### 4.52.3 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-

00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A “Data Butler” will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

## 4.53 LVV-T75 - Verify implementation of Multi-band Coadds (DMS-REQ-0281)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|

### 4.53.1 Test Items

Data Release Production

### 4.53.2 Requirements

- LVV-112 - DMS-REQ-0281-V-01: Multi-band Coadds

### 4.53.3 Precondition

### 4.53.4 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00)

**Step 5**
A “Data Butler” will be initialized to access the repository.

**Step 6**

For each combination of tract/patch/filter, the PVI will be retrieved from the Butler, and the existence of all components described in Test items section §4.6.2 will be verified.

**Step 7**
Scripts from the pipe_analysis package will be run on every visit to check for the presence of data products and make plots

**Step 8**
Ten patches will be chosen at random and inspected by eye for unmasked artifacts.

**Step 9**
Verify that deep detection coadds exist based on all filters.



## 4.54 LVV-T76 - Verify implementation of All-Sky Visualization of Data Releases (DMS-REQ-0329)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Simon Krughoff |

### 4.54.1 Test Items

Data Release Production

### 4.54.2 Requirements

- LVV-160 - DMS-REQ-0329-V-01: All-Sky Visualization of Data Releases

### 4.54.3 Precondition

Input Data
Dataset of perhaps ~100 square degrees. The first HSC Public Data Release will be used for

this test. Larger (in sky area) datasets should be identified for further testing.

### 4.54.4 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A "Data Butler" will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Run all sky tile generation task to produce the data products necessary for serving the all sky visualization.

**Step 5**
Manually perform, and log (including timing where applicable), the following steps against that all sky visualization application. At all steps take special care to note any missing or un-rendered image tiles:

1. Navigate to the all sky viewer and log the URL, browser and version.
2. Zoom to native pixel display (1 image pixel per display pixel)
3. Zoom to fit the full PDR footprint
4. Zoom to 1/4x native resolution
5. Pan to eastern edge of the footprint.
6. Pan to western edge of the footprint.
7. Navigate to the middle of the footprint.
8. Zoom to max magnification

## 4.55 LVV-T77 - Verify implementation of Best Seeing Coadds (DMS-REQ-0330)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.55.1 Test Items

Data Release Production

### 4.55.2 Requirements

- LVV-161 - DMS-REQ-0330-V-01: Best Seeing Coadds

### 4.55.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Explicitly create a coadd for a specified seeing range in each filter.

**Step 5**
Verify that these coadds exist.

## 4.56 LVV-T78 - Verify implementation of Persisting Data Products (DMS-REQ-0334)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.56.1 Test Items

Verify that per-band deep coadds and best-seeing coadds are present, kept, and available.

### 4.56.2 Requirements

- LVV-165 - DMS-REQ-0334-V-01: Persisting Data Products

### 4.56.3 Precondition

Precursor data from HSC PDR.

### 4.56.4 Test Script

**Step 1**

Produce some relevant coadds and store them in the Archive

**Step 2**
Examine the data retention policies for those products

## 4.57 LVV-T79 - Verify implementation of PSF-Matched Coadds (DMS-REQ-0335)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.57.1 Test Items

Data Release Production

### 4.57.2 Requirements

- LVV-166 - DMS-REQ-0335-V-01: PSF-Matched Coadds

### 4.57.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
Verify that PSF-matched coadds were created.

## 4.58 LVV-T80 - Verify implementation of Detecting faint variable objects (DMS-REQ-0337)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

### 4.58.1 Test Items

Data Release Production

### 4.58.2 Requirements

- LVV-168 - DMS-REQ-0337-V-01: Detecting faint variable objects

### 4.58.3 Precondition

Input Data
DECam HiTS data.
Gaia catalog of faint moving objects.
Catalog of spectroscopically confirmed quasars.
(Alternative: input data injected with faint variable sources).

### 4.58.4 Test Script

**Step 1**
The DM Stack and Alert Processing packaged shall be initialized as described in LVT-T17 (AG-00-00).

**Step 2**
The alert generation processing will be executed using the verification cluster:

‘‘‘bash
python ap_verify/bin/prepare_demo_slurm_files.py
# At present we must run a single ccd+visit to handle ingestion before
# parallel processing can begin
./ap_verify/bin/exec_demo_run_1ccd.sh 410915 25
ln -s ap_verify/bin/demo_run.sl
ln -s ap_verify/bin/demo_cmds.conf
sbatch demo_run.sl
‘‘‘

and any errors or failures reported.

**Step 3**
A "Data Butler" will be initialized to access the repository.

**Step 4**
For each of the expected data products types (listed in §4.2.2) and each of the expected units (PVIs, catalogs, etc.), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 5**
DIAObjects are currently only stored in a database, without shims to the Butler, so the existence of the database table and its non-empty contents will be verified by directly accessing it using sqlite3 and executing appropriate SQL queries.

**Step 6**
Identify 100 objects from Gaia with proper motions high enough to have detectably moved during HSC observations.

**Step 7**
Measure reported proper motion of these objects in DM Stack processing. Verify that it is consistent with Gaia objects.

**Step 8**
Identify 100 quasars from color-space or existing extragalactic spectroscopic catalog.

**Step 9**
Measure lightcurves of these quasars. Determine if structure function is reasonable (may require at least a year to determine if the structure function of 100 quasars is "reasonable").

**Step 10**
(Alternative: if faint variable source can be injected into the input data, test to see if they are recovered).

## 4.59 LVV-T81 - Verify implementation of Targeted Coadds (DMS-REQ-0338)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.59.1 Test Items

Remove DR from disk, observe retention of designated coadd sections, observe accessibility

### 4.59.2 Requirements

- LVV-169 - DMS-REQ-0338-V-01: Targeted Coadds

### 4.59.3 Test Script

**Step 1**
Remove DR from disk

**Step 2**
Observe retention of designated coadd sections

**Step 3**
Observe accessibility of designated coadd sections via simulated DAC LSP instance

## 4.60 LVV-T82 - Verify implementation of Tracking Characterization Changes Between Data Releases (DMS-REQ-0339)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Jim Bosch |

### 4.60.1 Test Items

Remove DR from disk, observe retention of designated catalog sections, observe accessibility

### 4.60.2 Requirements

- LVV-170 - DMS-REQ-0339-V-01: Tracking Characterization Changes Between Data Releases

### 4.60.3 Test Script

**Step 1**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 2**
A “Data Butler” will be initialized to access the repository.

**Step 3**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 4**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00).

**Step 5**
A “Data Butler” will be initialized to access the repository.

**Step 6**
Scripts from the pipe_analysis package will be run on every visit to check for the presence of data products and make plots.

**Step 7**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00).

**Step 8**
A "Data Butler" will be initialized to access the repository.

**Step 9**
Scripts from the pipe_analysis package will be run on every tract to check for the presence of data products and make plots

**Step 10**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00).

**Step 11**
A "Data Butler" will be initialized to access the repository.

**Step 12**
For each processed CCD, the PVI will be retrieved from the Butler, and the existence of all components described in section Test Items (§4.6.2) will be verified.

**Step 13**
Scripts from the pipe_analysis package will be run on every visit to check for the presence of data products and make plots

**Step 14**
Five sensors will be chosen at random from each of two visits and inspected by eye for unmasked artifacts.

**Step 15**
The DM Stack shall be initialized using the loadLSST script (as described in LVV-T10 - DRP-00-00)

**Step 16**
A "Data Butler" will be initialized to access the repository.

**Step 17**
For each combination of tract/patch/filter, the PVI will be retrieved from the Butler, and the

existence of all components described in Test items section §4.6.2 will be verified.

**Step 18**
Scripts from the pipe_analysis package will be run on every visit to check for the presence of data products and make plots

**Step 19**
Ten patches will be chosen at random and inspected by eye for unmasked artifacts.

**Step 20**
Prepare a second DRP run -> DPDD with different configuration parameters for this second test Data Release.

**Step 21**
Stage subset of products from first test Data Release to separate storage.

**Step 22**
Scientifically compare the results of the subset of that region of sky to those in the second test Data Release comparing the results of the DRP Scientific Verification tests.

## 4.61 LVV-T83 - Verify implementation of Bad Pixel Map (DMS-REQ-0059)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.61.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.61.2 Requirements

- LVV-22 - DMS-REQ-0059-V-01: Bad Pixel Map

### 4.61.3 Test Script

**Step 1**
Delegate to CPP

## 4.62 LVV-T84 - Verify implementation of Bias Residual Image (DMS-REQ-0060)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.62.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.62.2 Requirements

- LVV-23 - DMS-REQ-0060-V-01: Bias Residual Image

### 4.62.3 Test Script

**Step 1**
Delegate to CPP

## 4.63 LVV-T85 - Verify implementation of Crosstalk Correction Matrix (DMS-REQ-0061)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.63.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.63.2 Requirements

- LVV-24 - DMS-REQ-0061-V-01: Crosstalk Correction Matrix

### 4.63.3 Test Script

**Step 1**
Delegate to CPP

## 4.64 LVV-T86 - Verify implementation of Illumination Correction Frame (DMS-REQ-0062)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.64.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.64.2 Requirements

- LVV-25 - DMS-REQ-0062-V-01: Illumination Correction Frame

### 4.64.3 Test Script

**Step 1**
Delegate to CPP

## 4.65 LVV-T87 - Verify implementation of Monochromatic Flatfield Data Cube (DMS-REQ-0063)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.65.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.65.2 Requirements

- LVV-26 - DMS-REQ-0063-V-01: Monochromatic Flatfield Data Cube

### 4.65.3 Test Script

**Step 1**
Delegate to CPP

## 4.66 LVV-T88 - Verify implementation of Calibration Data Products (DMS-REQ-0130)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.66.1 Test Items

Data Backbone, Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.66.2 Requirements

- LVV-57 - DMS-REQ-0130-V-01: Calibration Data Products

### 4.66.3 Test Script

**Step 1**
Delegate to CPP

## 4.67 LVV-T89 - Verify implementation of Calibration Image Provenance (DMS-REQ-0132)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

### 4.67.1 Test Items

Batch Production, Managed Database, Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

### 4.67.2 Requirements

- LVV-59 - DMS-REQ-0132-V-01: Calibration Image Provenance

### 4.67.3 Test Script

**Step 1**
Precursor data, execute CPP, observe provenance

## 4.68 LVV-T90 - Verify implementation of Dark Current Correction Frame (DMS-REQ-0282)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

#### 4.68.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

#### 4.68.2 Requirements

- LVV-113 - DMS-REQ-0282-V-01: Dark Current Correction Frame

#### 4.68.3 Test Script

**Step 1**
Delegate to CPP

### 4.69 LVV-T91 - Verify implementation of Fringe Correction Frame (DMS-REQ-0283)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Lupton |

#### 4.69.1 Test Items

Daily Calibration Products Update, Periodic Calibration Products, Annual Calibration Products

#### 4.69.2 Requirements

- LVV-114 - DMS-REQ-0283-V-01: Fringe Correction Frame

### 4.69.3 Test Script

**Step 1**
Delegate to CPP

## 4.70 LVV-T92 - Verify implementation of Processing of Data From Special Programs (DMS-REQ-0320)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

### 4.70.1 Test Items

Special Programs Production, Data Release Production, Task Framework, Workload and Orchestration & Precursor data, execute representative Special Programs pipelines

### 4.70.2 Requirements

- LVV-151 - DMS-REQ-0320-V-01: Processing of Data From Special Programs

### 4.70.3 Precondition

A variety of imaging data from Special Programs, including these scenarios:
(1) Special Programs data that can be processed by the Prompt pipeline (i.e., standard visits)
(2) Special Programs data that requires 'real-time' (~24) processing with a reconfigured pipeline (e.g., DDF imaging sequence)
(3) Special Programs data that can (should) be processed by the Data Release pipeline (e.g., North Ecliptic Spur standard visits)

### 4.70.4 Test Script

**Step 1**
(1) Special Programs data that can be processed by the Prompt pipeline (i.e., standard visits).

Check that all images with the header keyword for SP were processed by the Prompt pipeline. Check that the Prompt pipeline's data products -- DIASource, DIAObject catalogs and the Alerts -- contain items flagged with their origin as that SP.

**Step 2**
(2) Special Programs data that requires 'real-time' (~24) processing with a reconfigured pipeline (e.g., DDF imaging sequence)
Check that all images with the header keywords for a given SP were processed by their re-configured pipeline. Check that the pipeline's data products have been updated, and passed their QA.

**Step 3**
(3) Special Programs data that can (should) be processed by the Data Release pipeline (e.g., North Ecliptic Spur standard visits).
SP data would be added manually to the DRP processing. Check that the DRP's data products -- Source, Object, CoAdds -- contain items flagged as originating in that SP.

## 4.71 LVV-T93 - Verify implementation of Level 1 Processing of Special Programs Data (DMS-REQ-0321)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

### 4.71.1 Test Items

Prompt Processing, Alert Production, MOPS and Forced Photometry & Execute single-day operations rehearsal, observe processing completed in time

### 4.71.2 Requirements

- LVV-152 - DMS-REQ-0321-V-01: Level 1 Processing of Special Programs Data

#### 4.71.3 Precondition

Imaging data obtained under a Special Program: for example, a sequence of consecutive images of a deep drilling field.

#### 4.71.4 Test Script

**Step 1**
If imaging data for a Special Program that requires processing with the Prompt pipeline was obtained the previous night, check that there exist DIASources/Objects/Alerts with flags that they originated from the Special Program.

**Step 2**
If imaging data for a Special Program that requires prompt processing with a reconfigured pipeline was obtained the previous night, check that the relevant data products have been updated.

### 4.72 LVV-T94 - Verify implementation of Special Programs Database (DMS-REQ-0322)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

#### 4.72.1 Test Items

Data Backbone, Managed Database, LSP Web APIs, Special Programs Production, Parallel Distributed Database & Precursor data, execute representative Special Programs pipelines, load results, observe distinct database

#### 4.72.2 Requirements

- LVV-153 - DMS-REQ-0322-V-01: Special Programs Database

### 4.72.3 Precondition

Databases created by reconfigured pipelines for processing Special Programs data (e.g., DIAObject/DIASource catalogs for a Deep Drilling Field)

### 4.72.4 Test Script

**Step 1**
SP data product: DDF DIAObjects catalog
Non-SP data product: WFD DIAObjects catalog
Test: join the two catalogs by coordinate (e.g., to get a longer time baseline for variable stars in the DDF)

**Step 2**
SP data product: DDF Objects catalog
Non-SP data product: WFD DIAObjects catalog
Test: join the two catalogs by coordinate to identify faint host galaxies of transients found in WFD

## 4.73 LVV-T95 - Verify implementation of Constraints on Level Special Program Products Generation (DMS-REQ-0344)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Melissa Graham |

### 4.73.1 Test Items

Execute single-day operations rehearsal, observe data products generated in time.

### 4.73.2 Requirements

- LVV-175 - DMS-REQ-0344-V-01: Constraints on Level 1 Special Program Products Generation

### 4.73.3 Precondition

Data from a Special Program that is appropriate for the Prompt pipeline (i.e., a standard visit from a non-crowded field).

### 4.73.4 Test Script

**Step 1**
Time processing of data starting from (pre-ingested) raw files until an alert is available for distribution; verify that this time is less than OTT1.

**Step 2**
Time processing of data starting from (pre-ingested) raw files until the required data products are available in the Science Platform. Verify that this time is less than L1PublicT.

**Step 3**
Run MOPS on 1 night equivalent of LSST observing worth of precursor data and verify that Solar System Object orbits can be updated within 24 hours.

**Step 4**

**Step 5**

**Step 6**
Record time between completion of MOPS processing and availability of the updated SSObject catalogue through the Science Platform; verify this time is less than L1PublicT.

## 4.74 LVV-T96 - Verify implementation of Query Repeatability (DMS-REQ-0291)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.74.1 Test Items

Load multiple DRs and PPDB, observe repeatability

Data Backbone, Managed Database, LSP Web APIs, Parallel Distributed Database

### 4.74.2 Requirements

- LVV-122 - DMS-REQ-0291-V-01: Query Repeatability

### 4.74.3 Test Script

**Step 1**
Select and download (deterministic) random subsample of records from Data Release Object and Source tables.

**Step 2**
Select and download random subsample of PPDB DIAObject and DIASource tables.

**Step 3**
As appropriate, wait for some amount of non-trivial database usage to occur, such as Prompt Processing ingestion or ingestion of other DRP database tables.

**Step 4**
Re-run the queries in steps 1 and 2 and verify that the resulting data are identical.

## 4.75 LVV-T97 - Verify implementation of Uniqueness of IDs Across Data Releases (DMS-REQ-0292)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.75.1 Test Items

Load multiple DRs and PPDB, observe uniqueness of IDs

### 4.75.2 Requirements

- LVV-123 - DMS-REQ-0292-V-01: Uniqueness of IDs Across Data Releases

### 4.75.3 Test Script

**Step 1**
Load multiple DRs and PPDB

**Step 2**
Observe uniqueness of IDs

## 4.76 LVV-T98 - Verify implementation of Selection of Datasets (DMS-REQ-0293)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.76.1 Test Items

Load DR, observe retrieval of representative datasets

### 4.76.2 Requirements

- LVV-124 - DMS-REQ-0293-V-01: Selection of Datasets

### 4.76.3 Test Script

**Step 1**
Load DR

**Step 2**
Observe retrieval of single PVI with metadata

**Step 3**
Observe retrieval of multiple PVIs with metadata

**Step 4**
Observe retrieval of coadd patch with metadata

**Step 5**
Observe retrieval of subset of rows in each catalog

## 4.77 LVV-T99 - Verify implementation of Processing of Datasets (DMS-REQ-0294)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.77.1 Test Items

Execute AP and DRP, simulate failures, observe correct processing

### 4.77.2 Requirements

- LVV-125 - DMS-REQ-0294-V-01: Processing of Datasets

### 4.77.3 Test Script

**Step 1**
Execute AP and DRP

**Step 2**
Simulate failures

**Step 3**
Observe correct processing

## 4.78 LVV-T100 - Verify implementation of Transparent Data Access (DMS-REQ-0295)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.78.1 Test Items

Observe dataset retrieval from multiple LSP instances

### 4.78.2 Requirements

- LVV-126 - DMS-REQ-0295-V-01: Transparent Data Access

### 4.78.3 Test Script

**Step 1**
Observe dataset retrieval from multiple LSP instances

## 4.79 LVV-T101 - Verify implementation of Transient Alert Distribution (DMS-REQ-0002)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.79.1 Test Items

Precursor or simulated data, execute AP, observe distribution to simulated clients using standard protocols

### 4.79.2 Requirements

- LVV-3 - DMS-REQ-0002-V-01: Transient Alert Distribution

### 4.79.3 Precondition

Obtain precursor or simulated data

### 4.79.4 Test Script

**Step 1**
Execute AP

**Step 2**
Observe distribution to simulated clients using standard protocols

## 4.80 LVV-T102 - Verify implementation of Solar System Objects Available Within Specified Time (DMS-REQ-0089)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.80.1 Test Items

Execute single-day operations rehearsal, observe data products generated in time

### 4.80.2 Requirements

- LVV-36 - DMS-REQ-0089-V-01: Solar System Objects Available Within Specified Time

### 4.80.3 Test Script

**Step 1**
Execute single-day operations rehearsal

**Step 2**
Observe data products generated in time

## 4.81 LVV-T103 - Verify implementation of Generate Data Quality Report Within Specified Time (DMS-REQ-0096)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.81.1 Test Items

Quality Control System

### 4.81.2 Requirements

- LVV-38 - DMS-REQ-0096-V-01: Generate Data Quality Report Within Specified Time

### 4.81.3 Test Script

**Step 1**
Execute single-day operations rehearsal

**Step 2**
Observe data quality report is generated on time and with correct contents

## 4.82 LVV-T104 - Verify implementation of Generate DMS Performance Report Within Specified Time (DMS-REQ-0098)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.82.1 Test Items

Quality Control System

### 4.82.2 Requirements

- LVV-40 - DMS-REQ-0098-V-01: Generate DMS Performance Report Within Specified Time

### 4.82.3 Test Script

**Step 1**
Execute single-day operations rehearsal

**Step 2**
Observe performance report is generated on time and with correct contents

## 4.83 LVV-T105 - Verify implementation of Generate Calibration Report Within Specified Time (DMS-REQ-0100)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.83.1 Test Items

QC System, Daily Calibration Products Update

### 4.83.2 Requirements

- LVV-42 - DMS-REQ-0100-V-01: Generate Calibration Report Within Specified Time

### 4.83.3 Test Script

**Step 1**
Execute single-day operations rehearsal

**Step 2**
Observe calibration report is generated on time and with correct contents

## 4.84 LVV-T106 - Verify implementation of Calibration Images Available Within Specified Time (DMS-REQ-0131)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.84.1 Test Items

Execute single-day operations rehearsal, observe data products generated

### 4.84.2 Requirements

- LVV-58 - DMS-REQ-0131-V-01: Calibration Images Available Within Specified Time

### 4.84.3 Test Script

**Step 1**
Execute single-day operations rehearsal

**Step 2**
Observe data products generated

## 4.85 LVV-T107 - Verify implementation of Level-1 Production Completeness (DMS-REQ-0284)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.85.1 Test Items

Image and EFD Archiving, Prompt Processing, Observatory Operations Data

### 4.85.2 Requirements

- LVV-115 - DMS-REQ-0284-V-01: Level-1 Production Completeness

### 4.85.3 Test Script

**Step 1**
Ingest raw data while simulating failures and outages, observe eventual recovery

## 4.86 LVV-T108 - Verify implementation of Level 1 Source Association (DMS-REQ-0285)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.86.1 Test Items

Alert Production

### 4.86.2 Requirements

- LVV-116 - DMS-REQ-0285-V-01: Level 1 Source Association

### 4.86.3 Test Script

**Step 1**
Delegate to AP

## 4.87 LVV-T109 - Verify implementation of SSObject Precovery (DMS-REQ-0286)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.87.1 Test Items

MOPS and Forced Photometry

### 4.87.2 Requirements

- LVV-117 - DMS-REQ-0286-V-01: SSObject Precovery

### 4.87.3 Test Script

**Step 1**
Delegate to AP

## 4.88 LVV-T110 - Verify implementation of DIASource Precovery (DMS-REQ-0287)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.88.1 Test Items

LSP Web APIs, MOPS and Forced Photometry

### 4.88.2 Requirements

- LVV-118 - DMS-REQ-0287-V-01: DIASource Precovery

### 4.88.3 Test Script

**Step 1**
Execute single-day operations rehearsal, observe data products generated in time

## 4.89 LVV-T111 - Verify implementation of Use of External Orbit Catalogs (DMS-REQ-0288)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.89.1 Test Items

Prompt Processing, Alert Production, MOPS and Forced Photometry

### 4.89.2 Requirements

- LVV-119 - DMS-REQ-0288-V-01: Use of External Orbit Catalogs

### 4.89.3 Test Script

**Step 1**
Delegate to AP

## 4.90 LVV-T112 - Verify implementation of Alert Filtering Service (DMS-REQ-0342)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.90.1 Test Items

Alert Filtering, LSP Portal

### 4.90.2 Requirements

- LVV-173 - DMS-REQ-0342-V-01: Alert Filtering Service

### 4.90.3 Test Script

**Step 1**
Simulated alert stream, observe ability to define filters and proper filter results

## 4.91 LVV-T113 - Verify implementation of Performance Requirements for LSST Alert Filtering Service (DMS-REQ-0343)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.91.1 Test Items

Alert Distribution, Alert Filtering, LSP Portal

### 4.91.2 Requirements

- LVV-174 - DMS-REQ-0343-V-01: Performance Requirements for LSST Alert Filtering Service

### 4.91.3 Test Script

**Step 1**
Simulated alert stream, observe ability to support specified load

## 4.92 LVV-T114 - Verify implementation of Pre-defined alert filters (DMS-REQ-0348)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Eric Bellm |

### 4.92.1 Test Items

Alert Filtering, LSP Portal

#### 4.92.2 Requirements

- LVV-179 - DMS-REQ-0348-V-01: Pre-defined alert filters

#### 4.92.3 Test Script

**Step 1**
Simulated alert stream, observe predefined filter existence and proper filter results

### 4.93 LVV-T115 - Verify implementation of Calibration Production Processing (DMS-REQ-0289)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

#### 4.93.1 Test Items

Execute CPP on a variety of representative cadences

#### 4.93.2 Requirements

- LVV-120 - DMS-REQ-0289-V-01: Calibration Production Processing

#### 4.93.3 Test Script

**Step 1**
Execute CPP on a variety of representative cadences

**Step 2**
Observe lack of failures and expected data products

## 4.94 LVV-T116 - Verify implementation of Associating Objects across data releases (DMS-REQ-0350)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.94.1 Test Items

Load DR, observe queryable association

### 4.94.2 Requirements

- LVV-181 - DMS-REQ-0350-V-01: Associating Objects across data releases

### 4.94.3 Test Script

**Step 1**
Load DR

**Step 2**
Observe queryable association

## 4.95 LVV-T117 - Verify implementation of DAC resource allocation for Level 3 processing (DMS-REQ-0119)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.95.1 Test Items

Observe resource allocation in PDAC

Batch Computing, Containerized Application Management, Identity Management, LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.95.2 Requirements

- LVV-47 - DMS-REQ-0119-V-01: DAC resource allocation for Level 3 processing

### 4.95.3 Test Script

**Step 1**
Create a test user account for the Science Platform.

**Step 2**
Set the LSP resource allocations for the test user to very low values.

**Step 3**
Initiate example batch jobs and notebook sessions that will exceed the specified resource limits.

**Step 4**
Transfer sufficient data volumes into the user workspace and MyDB tables that would exceed the resource quotas.

**Step 5**
Reset the user resource quotas to normal values.

**Step 6**
Initiate the same example batch jobs and notebook sessions that previously caused an error.

**Step 7**
Transfer the same data volumes into the user workspace and MyDB tables that previously caused an error.

## 4.96 LVV-T118 - Verify implementation of Level 3 Data Product Self Consistency (DMS-REQ-0120)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.96.1 Test Items

Data Backbone, LSP Web APIs, Workload and Orchestration

### 4.96.2 Requirements

- LVV-48 - DMS-REQ-0120-V-01: Level 3 Data Product Self Consistency

### 4.96.3 Test Script

**Step 1**
Execute representative processing on DR in PDAC, observe consistency

## 4.97 LVV-T119 - Verify implementation of Provenance for Level 3 processing at DACs (DMS-REQ-0121)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.97.1 Test Items

Data Backbone, LSP Web APIs, Data Butler Access Client, Task Framework, Workload and Orchestration

### 4.97.2 Requirements

- LVV-49 - DMS-REQ-0121-V-01: Provenance for Level 3 processing at DACs

### 4.97.3 Test Script

**Step 1**
Execute representative processing on DR in PDAC, observe provenance recording

## 4.98 LVV-T120 - Verify implementation of Software framework for Level 3 catalog processing (DMS-REQ-0125)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.98.1 Test Items

Data Backbone, LSP JupyterLab, LSP Web APIs, Data Butler Access Client, Task Framework, Workload and Orchestration

### 4.98.2 Requirements

- LVV-53 - DMS-REQ-0125-V-01: Software framework for Level 3 catalog processing

### 4.98.3 Test Script

**Step 1**
Execute representative processing on DR in PDAC, observe recognition of and recovery from failures

## 4.99 LVV-T121 - Verify implementation of Software framework for Level 3 image processing (DMS-REQ-0128)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.99.1 Test Items

Data Backbone, LSP JupyterLab, LSP Web APIs, Data Butler Access Client, Task Framework, Workload and Orchestration

### 4.99.2 Requirements

- LVV-56 - DMS-REQ-0128-V-01: Software framework for Level 3 image processing

### 4.99.3 Test Script

**Step 1**
Execute representative processing on DR in PDAC, observe recognition of and recovery from failures

## 4.100 LVV-T122 - Verify implementation of Level 3 Data Import (DMS-REQ-0290)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.100.1 Test Items

Load representative files into PDAC

Data Backbone, LSP Web APIs, Parallel Distributed Database

### 4.100.2 Requirements

- LVV-121 - DMS-REQ-0290-V-01: Level 3 Data Import

### 4.100.3 Test Script

**Step 1**
Use the Science Platform catalog upload tool to ingest a small example FITS table.

**Step 2**
Use the Science Platform catalog upload tool to ingest a small example CSV table.

**Step 3**
Use the Science Platform catalog upload tool to ingest a large FITS table that needs to be spatially-sharded in the database.

**Step 4**
Perform example queries on each of the three tables to verify that all data is present.

## 4.101 LVV-T123 - Verify implementation of Access Controls of Level 3 Data Products (DMS-REQ-0340)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.101.1 Test Items

Data Backbone, IT Security, Identity Management, LSP Portal, Parallel Distributed Database

### 4.101.2 Requirements

- LVV-171 - DMS-REQ-0340-V-01: Access Controls of Level 3 Data Products

#### 4.101.3 Test Script

**Step 1**
Configure representative access controls in PDAC, observe proper restrictions

### 4.102 LVV-T124 - Verify implementation of Software Architecture to Enable Community Re-Use (DMS-REQ-0308)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Simon Krughoff |

#### 4.102.1 Test Items

Execution of algorithms on batch cluster and desktop.

#### 4.102.2 Requirements

- LVV-139 - DMS-REQ-0308-V-01: Software Architecture to Enable Community Re-Use

#### 4.102.3 Test Script

**Step 1**
Using curated test datasets for multiple precursor instruments, verify and log that the prototype DRP pipelines execute successfully in three contexts:
1. The CI system
2. On a single user system: laptop, desktop, or notebook running in the Notebook aspect of the LSP.

**Step 2**
Using a template testing notebook in the Notebook aspect of the LSP, verify and log the following:
1. Individual pipeline steps (tasks) are importable and executable on their own. this is not comprehensive, but demonstrative.

2. Individual pipeline steps may be overridden by configuration.
3. Users can implement a custom pipeline step and insert i into the processing flow via configuration.

**Step 3**
The DM Stack shall be initialized using the loadLSST script (as described in DRP-00-00).

**Step 4**
A "Data Butler" will be initialized to access the repository.

**Step 5**
For each of the expected data products types (listed in Test Items section §4.3.2) and each of the expected units (PVIs, coadds, etc), the data product will be retrieved from the Butler and verified to be non-empty.

**Step 6**
Run subset of full DRP from previous step on an individual node. Was this organizationally easy? Did the performance scale appropriately?

**Step 7**
Re-run aperture correction on subset. Verify that same results as DRP run are achieved.

**Step 8**
Re-run photometric redshift estimation algorithm on subset coadd catalogs. Verify that same results are achieved as from full DRP.

### 4.103 LVV-T125 - Verify implementation of Simulated Data (DMS-REQ-0009)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.103.1 Test Items

Alert Production, Data Release Production

### 4.103.2 Requirements

- LVV-6 - DMS-REQ-0009-V-01: Simulated Data

### 4.103.3 Test Script

**Step 1**
Delegate to AP and DRP

## 4.104 LVV-T126 - Verify implementation Image Differencing (DMS-REQ-0032)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.104.1 Test Items

Alert Production, Data Release Production, Science Algorithms

### 4.104.2 Requirements

- LVV-14 - DMS-REQ-0032-V-01: Image Differencing

### 4.104.3 Test Script

**Step 1**
Delegate to AP and DRP

## 4.105 LVV-T127 - Verify implementation of Provide Source Detection Software (DMS-REQ-0033)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.105.1 Test Items

Alert Production, Data Release Production, Science Algorithms

### 4.105.2 Requirements

- LVV-15 - DMS-REQ-0033-V-01: Provide Source Detection Software

### 4.105.3 Test Script

**Step 1**
Delegate to AP and DRP

## 4.106 LVV-T128 - Verify implementation Provide Astrometric Model (DMS-REQ-0042)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.106.1 Test Items

Alert Production, Data Release Production, Science Algorithms, Science Primitives

### 4.106.2 Requirements

- LVV-17 - DMS-REQ-0042-V-01: Provide Astrometric Model

#### 4.106.3 Test Script

**Step 1**
Delegate to AP and DRP

### 4.107 LVV-T129 - Verify implementation of Provide Calibrated Photometry (DMS-REQ-0043)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.107.1 Test Items

Alert Production, Data Release Production, Science Algorithms, Science Primitives

#### 4.107.2 Requirements

- LVV-18 - DMS-REQ-0043-V-01: Provide Calibrated Photometry

#### 4.107.3 Test Script

**Step 1**
Delegate to AP and DRP

### 4.108 LVV-T130 - Verify implementation of Enable a Range of Shape Measurement Approaches (DMS-REQ-0052)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.108.1 Test Items

Alert Production, Data Release Production, Science Algorithms, Science Primitives

### 4.108.2 Requirements

- LVV-21 - DMS-REQ-0052-V-01: Enable a Range of Shape Measurement Approaches

### 4.108.3 Test Script

**Step 1**
Delegate to AP and DRP

## 4.109 LVV-T131 - Verify implementation of Provide User Interface Services (DMS-REQ-0160)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Gregory Dubois-Felsmann |

### 4.109.1 Test Items

Delegate to LSP: it is intended that the detailed test coverage for this requirement will largely arise from testing at the Science Platform level.

### 4.109.2 Requirements

- LVV-63 - DMS-REQ-0160-V-01: Provide User Interface Services

### 4.109.3 Precondition

1. Testing this requirement relies on a set of data products meeting the data model implied by the DPDD existing in a deployment of the Science Platform and its underlying database and file services.

(a) In particular, both image and catalog data products are required.

(b) From the specific language of the underlying requirement, it appears clear that coadded data products are required, but in practice single-epoch data products should be included in the test as well.

2. Depending on when this requirement is tested, the tests may involve either or both of precursor data and LSST commissioning data. The use of the latter is ultimately essential to ensure that the tests are performed with as LSST-like a dataset as possible.

#### 4.109.4 Test Script

**Step 1**

**Establishment of test coordinates:**

Establish sky positions and surrounding regions (e.g., cones or polygons), field sizes, filter bands, and temporal epochs for the tests that are consistent with the known content of the test dataset, whether precursor or LSST commissioning data.

Establishing sky positions should include pre-determining the corresponding LSST "tract and patch" identifiers.

If the plan to not keep all calibrated single-epoch images on disk is still in place at the time of the test, identify for use in the test both images that are, and are not, on disk.

Establish target image boundaries, projections, and pixel scales to be used for resampling tests. Ensure that at least some of these test conditions include coadded image boundaries that cross tract and patch boundaries, and single-epoch image boundaries that cross focal plane raft boundaries.

**Step 2**

**Butler image access:**

From within the Notebook Aspect, verify that coadded images for the identified regions of sky and filter bands are accessible via the Butler. Verify that the same images are available whether obtained by direct reference to the previous established tract/patch identifiers or by the use of LSST stack code for retrieving images based on sky coordinates.

From within the Notebook Aspect, verify that single-epoch raw images for the selected locations and times are available. Verify that calibrated images (PVIs) for the selected locations and times are available; depending on the details of the test dataset, verify that PVIs still on disk can be retrieved immediately.

Verify that lists or tables of image metadata, not just individual images, can be retrieved. E.g., a list of all the single-epoch images covering a selected sky location.

**Step 3**

**Programmatic PVI re-creation:**

From within the Notebook Aspect, verify that the recreation on demand of a PVI can be performed. Ideally, this should be done as follows:

- Verify that recreation of a PVI that *is* still available works and that it reproduces the original PVI exactly (except for provenance metadata that must be different) or within the reasonable ability of processing systems to do so (e.g., taking into account that the original calibration and the recreation may have run on different CPU architectures).
- The test conditions should ensure the verification that a recreation was actually performed, i.e., that the still-available PVI was not returned instead.
- Note that it does not appear to be a requirement that *at Butler level* recreation on demand of PVIs is a completely transparent process. If this *is* decided to be a requirement, the test must also verify that it has been satisfied. If it is *not* a requirement, verify that adequate documentation on the PVI-recreation process (e.g., the SuperTasks and configuration to be used) is available.

**Step 4**

**Butler catalog access:**

From within the Notebook Aspect, verify that all the catalog data products described in the DPDD can be retrieved for the coordinates selected above via the Butler. (This test should include access to SSObject data, but the details of how such a test would depend on the coordinate selections require additional thought.)

**Step 5**

**LSST-stack-based resampling/reprojection:**

Verify the availability of software in the LSST stack, and associated documentation, that permits the resampling of LSST images to different pixel grids and projections.
Exercise this capability for the test conditions selected in Step 1 above.
Perform photometric and astrometric tests on the resulting resampled images to provide evidence that the transformations performed were correct to the accuracy supported by the data.

**Step 6**

**Comment:**

The following API Aspect test steps should be carried out on the required "offsite-like" test platform, to ensure that their success does not reflect any privileged access given to processes inside the Data Access Center or other Science Platform instance. However, at least a small sampling of them should *also* be carried out *within* the Science Platform environment, i.e., in the Notebook Aspect, and the results compared.

**Step 7**

**API Aspect image access:**

Using IVOA services such as the Registry and ObsTAP, from the "offsite-like" test platform, verify that the existence of the classes of image data products foreseen in the DPDD can be determined.

Verify that ObsTAP and/or SIAv2 can be used to find the same images and lists of images for the established test coordinates that were retrieved via the Butler in Step 2 above.

Verify that the selected images are retrievable from the Web services.

Verify that the retrieved images are identical in their pixel content and metadata.

The tests must include both coadded and single-epoch images.

**Step 8**

**API Aspect image transformations:**

Verify that image cutouts and resamplings can be performed via the IVOA SODA service, and that the results are identical to those obtained for the same parameters from the LSST-stack-based tests in Step 5.

(The requirements for supported reprojections, if any, in the SODA service have not been established at the time of writing.)

**Step 9**

**API Aspect catalog data access:**

Verify that the IVOA Registry, RegTAP, TAP_SCHEMA, and other relevant mechanisms can be used to discover the existence of all the catalog data products foreseen in the DPDD.

Using the IVOA TAP service, verify that all the catalog data products foreseen in the DPDD can be retrieved for the coordinates determined in Step 1. Verify that their scientific content is the same as when they are retrieved via the Butler.

**Step 10**

**Comment:**

The Portal Aspect tests below should be carried out from a web browser on an "offsite-like"

test platform, to ensure that no privileged access provided to intra-data-center clients is relied upon.

**Step 11**
**Portal Aspect data browsing:**
Verify that the Portal Aspect can be used to discover the existence of all the data products foreseen in the DPDD. Verify that the UI permits locating the data for the coordinates selected in Step 1 by visual means, e.g., by zooming and panning in from an all-sky view.
Verify that the UI permits locating the data by typing in coordinates as well.

**Step 12**
**Portal Aspect image access:**
Verify that the Portal Aspect allows both the retrieval of "original" image data, i.e., in its native LSST pixel projection and with full metadata, as well as retrieval of on-demand UI cutouts of coadded image data for selected locations.

**Step 13**
**Portal Aspect catalog query and visualization:**
Verify that the Portal Aspect allows graphical querying of DPDD catalog data, both coadded and single-epoch, for selected regions of sky and/or with selected properties, and supports the visualization of the results (including histogramming, scatterplots, time series, table manipulations, and overplotting on image data).
(Note that the Science Platform requirements, LDM-554, lay out a detailed set of requirements on the selection and visualization of catalog data.)

**Step 14**
**Portal Aspect data download:**
Verify that data identified and/or visualized in the Portal Aspect can be downloaded to the remote system running the web browser in which the Portal is displayed, as well as to the User Workspace.

### 4.110 LVV-T132 - Verify implementation of Pre-cursor, and Real Data (DMS-REQ-0296)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.110.1 Test Items

Science Algorithms, Data Butler Access Client

#### 4.110.2 Requirements

- LVV-127 - DMS-REQ-0296-V-01: Pre-cursor, and Real Data

#### 4.110.3 Test Script

**Step 1**
Execute AP and DRP on non-LSST data

### 4.111 LVV-T133 - Verify implementation of Provide Beam Projector Coordinate Calculation Software (DMS-REQ-0351)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.111.1 Test Items

Science Primitives

#### 4.111.2 Requirements

- LVV-182 - DMS-REQ-0351-V-01: Provide Beam Projector Coordinate Calculation Software

#### 4.111.3 Test Script

**Step 1**
Delegate to CPP

### 4.112 LVV-T134 - Verify implementatino of Provide Image Access Services (DMS-REQ-0065)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.112.1 Test Items

LSP Portal, LSP Web APIs

#### 4.112.2 Requirements

- LVV-27 - DMS-REQ-0065-V-01: Provide Image Access Services

#### 4.112.3 Test Script

**Step 1**
Delegate to LSP

### 4.113 LVV-T135 - Verify implementation of Provide Data Access Services (DMS-REQ-0155)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.113.1 Test Items

LSP Web APIs

### 4.113.2 Requirements

- LVV-60 - DMS-REQ-0155-V-01: Provide Data Access Services

### 4.113.3 Test Script

**Step 1**
Delegate to LSP

## 4.114 LVV-T136 - Verify implementation of Data Product and Raw Data Access (DMS-REQ-0298)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.114.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP Web APIs

### 4.114.2 Requirements

- LVV-129 - DMS-REQ-0298-V-01: Data Product and Raw Data Access

### 4.114.3 Test Script

**Step 1**
Delegate to LSP

## 4.115 LVV-T137 - Verify implementation of Data Product Ingest (DMS-REQ-0299)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.115.1 Test Items

Data Backbone, LSP Web APIs

### 4.115.2 Requirements

- LVV-130 - DMS-REQ-0299-V-01: Data Product Ingest

### 4.115.3 Test Script

**Step 1**
Delegate to DBB

## 4.116 LVV-T138 - Verify implementation Bulk Download Service (DMS-REQ-0300)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

### 4.116.1 Test Items

Bulk Download

### 4.116.2 Requirements

- LVV-131 - DMS-REQ-0300-V-01: Bulk Download Service

### 4.116.3 Precondition

A large dataset (at least a few TB) must be available.
Requires identity management to confirm bulk download use.
While this can be tested and shown to work using LSST DAC, Chilean DAC, and IN2P3 endpoints, this should also be tested to demonstrate expected throughput for outside users (e.g. FNAL, NERSC sites could be tested).

### 4.116.4 Test Script

**Step 1**
Setup large transfer request and examine the data transfer rates achieved.

**Step 2**
Test should be repeated while observing in firehose mode (with LSSTCam) during science verification to ensure that bulk transfer does not compromise normal nightly operations.

## 4.117 LVV-T139 - Verify implementation of Provide Pipeline Execution Services (DMS-REQ-0156)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.117.1 Test Items

Batch Production, Workload and Orchestration

### 4.117.2 Requirements

- LVV-61 - DMS-REQ-0156-V-01: Provide Pipeline Execution Services

#### 4.117.3 Test Script

**Step 1**
Verify subsidiary requirements

### 4.118 LVV-T140 - Verify implementation of Production Orchestration (DMS-REQ-0302)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.118.1 Test Items

Batch Production, Workload and Orchestration

#### 4.118.2 Requirements

- LVV-133 - DMS-REQ-0302-V-01: Production Orchestration

#### 4.118.3 Test Script

**Step 1**
Delegate to Batch Production

### 4.119 LVV-T141 - Verify implementation of Production Monitoring ( DMS-REQ-0303)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.119.1 Test Items

Service Management and Monitoring, Workload and Orchestration

#### 4.119.2 Requirements

- LVV-134 - DMS-REQ-0303-V-01: Production Monitoring

#### 4.119.3 Test Script

**Step 1**
Observe monitoring during DRP execution

### 4.120 LVV-T142 - Verify implementation of Production Fault Tolerance (DMS-REQ-0304)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.120.1 Test Items

Batch Production, Task Framework, Workload and Orchestration

#### 4.120.2 Requirements

- LVV-135 - DMS-REQ-0304-V-01: Production Fault Tolerance

#### 4.120.3 Test Script

**Step 1**
Execute AP and DRP, simulate failures, observe correct processing

## 4.121 LVV-T143 - Verify implementation of Provide Pipeline Construction Services (DMS-REQ-0158)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.121.1 Test Items

Task Framework

### 4.121.2 Requirements

- LVV-62 - DMS-REQ-0158-V-01: Provide Pipeline Construction Services

### 4.121.3 Test Script

**Step 1**
Delegate to Middleware

## 4.122 LVV-T144 - Verify implementation of Task Specification (DMS-REQ-0305)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.122.1 Test Items

Middleware

Task Framework

### 4.122.2 Requirements

- LVV-136 - DMS-REQ-0305-V-01: Task Specification

### 4.122.3 Test Script

**Step 1**
Inspect software architecture. Verify that there exists Tasks that can be run and configured without re-complication.

**Step 2**
Verify that an example science algorithm can be run through one of these Tasks. Three examples from different areas: source measurement, image subtraction, and photometric-redshift estimation.

## 4.123 LVV-T145 - Verify implementation of Task Configuration (DMS-REQ-0306)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.123.1 Test Items

Task Framework

### 4.123.2 Requirements

- LVV-137 - DMS-REQ-0306-V-01: Task Configuration

### 4.123.3 Test Script

**Step 1**
Inspect software design to verify that one can define the configuration for a Task.

**Step 2**

Run a Task with a known invalid configuration. Verify that the error is caught before the science algorithm executes.

**Step 3**

Run a simple task with two different configurations that make a material difference for a Task. E.g., specify a different source detection threshold. Verify that the configuration is different between the two runs through difference in recorded provenance and in results.

## 4.124 LVV-T146 - Verify implementation of DMS Initialization Component (DMS-REQ-0297)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.124.1 Test Items

Compute/Storage/LAN

### 4.124.2 Requirements

- LVV-128 - DMS-REQ-0297-V-01: DMS Initialization Component

### 4.124.3 Test Script

**Step 1**

Power-cycle all DM systems at each Facility, observe recovery

## 4.125 LVV-T147 - Verify implementation pof Control of Level-1 Production (DMS-REQ-0301)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.125.1 Test Items

Image and EFD Archiving, Pointing Prediction Publishing, Prompt Processing, OCS Driven Batch

#### 4.125.2 Requirements

- LVV-132 - DMS-REQ-0301-V-01: Control of Level-1 Production

#### 4.125.3 Test Script

**Step 1**
Observe existence and capability of Prompt DMCS

### 4.126 LVV-T148 - Verify implementation of Unique Processing Coverage (DMS-REQ-0307)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.126.1 Test Items

Batch Production, Workload and Orchestration

#### 4.126.2 Requirements

- LVV-138 - DMS-REQ-0307-V-01: Unique Processing Coverage

### 4.126.3 Test Script

**Step 1**
Execute representative processing, observe lack of duplicates or missing rows even in the presence of failures

## 4.127 LVV-T149 - Verify implementation of Catalog Queries (DMS-REQ-0075)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.127.1 Test Items

Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.127.2 Requirements

- LVV-33 - DMS-REQ-0075-V-01: Catalog Queries

### 4.127.3 Test Script

**Step 1**
Delegate to LSP

## 4.128 LVV-T150 - Verify implementation of Maintain Archive Publicly Accessible (DMS-REQ-0077)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.128.1 Test Items

Data Backbone, Managed Database, Service Management and Monitoring, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.128.2 Requirements

- LVV-34 - DMS-REQ-0077-V-01: Maintain Archive Publicly Accessible

### 4.128.3 Test Script

**Step 1**
Observe access to prior DR on tape

## 4.129 LVV-T151 - Verify implementation of Catalog Export Formats (DMS-REQ-0078)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

### 4.129.1 Test Items

LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.129.2 Requirements

- LVV-35 - DMS-REQ-0078-V-01: Catalog Export Formats

### 4.129.3 Test Script

**Step 1**
Delegate to LSP

## 4.130 LVV-T152 - Verify implementation of Keep Historical Alert Archive (DMS-REQ-0094)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.130.1 Test Items

Data Backbone, LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.130.2 Requirements

- LVV-37 - DMS-REQ-0094-V-01: Keep Historical Alert Archive

### 4.130.3 Test Script

**Step 1**
Simulated alert stream, load Alert DB, observe access to Alert DB

## 4.131 LVV-T153 - Verify implementation of Provide Engineering and Facility Database Archive (DMS-REQ-0102)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.131.1 Test Items

Image and EFD Archiving, Data Backbone, Managed Database

#### 4.131.2 Requirements

- LVV-44 - DMS-REQ-0102-V-01: Provide Engineering & Facility Database Archive

#### 4.131.3 Test Script

**Step 1**
Execute single-day operations rehearsal, observe data products generated in time

### 4.132 LVV-T154 - Verify implementation of Raw Data Archiving Reliability (DMS-REQ-0309)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.132.1 Test Items

Image and EFD Archiving, Data Backbone, Managed Database

#### 4.132.2 Requirements

- LVV-140 - DMS-REQ-0309-V-01: Raw Data Archiving Reliability

#### 4.132.3 Test Script

**Step 1**
Analyze sources of loss or corruption after mitigation to compute estimated reliability

### 4.133 LVV-T155 - Verify implementation of Un-Archived Data Product Cache (DMS-REQ-0310)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

#### 4.133.1 Test Items

Data Backbone

#### 4.133.2 Requirements

- LVV-141 - DMS-REQ-0310-V-01: Un-Archived Data Product Cache

#### 4.133.3 Test Script

**Step 1**
Delegate to DBB

### 4.134 LVV-T156 - Verify implementation of Regenerate Un-archived Data Products (DMS-REQ-0311)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

#### 4.134.1 Test Items

Science Portal (note: here and below, LSP test must be without shims for DBB)

#### 4.134.2 Requirements

- LVV-142 - DMS-REQ-0311-V-01: Regenerate Un-archived Data Products

#### 4.134.3 Test Script

**Step 1**
Run a small DRP processing job and download unarchived data products.

**Step 2**
Wait for (or force) a processing stack change so that the subsequent re-processing will be forced to use an older software build.

**Step 3**
Using provenance information from the products in Step 1, request a re-processing and compare results with previously unarchived products.

### 4.135 LVV-T157 - Verify implementation Level 1 Data Product Access (DMS-REQ-0312)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.135.1 Test Items

Prompt Processing, Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Alert Production

#### 4.135.2 Requirements

- LVV-143 - DMS-REQ-0312-V-01: Level 1 Data Product Access

#### 4.135.3 Test Script

**Step 1**
Delegate to LSP

## 4.136 LVV-T158 - Verify implementation Level 1 and 2 Catalog Access (DMS-REQ-0313)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.136.1 Test Items

Data Backbone, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.136.2 Requirements

- LVV-144 - DMS-REQ-0313-V-01: Level 1 & 2 Catalog Access

### 4.136.3 Test Script

**Step 1**
Delegate to LSP

## 4.137 LVV-T159 - Verify implementation of Regenerating Data Products from Previous Data Releases (DMS-REQ-0336)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

### 4.137.1 Test Items

Data Backbone, LSP Web APIs

### 4.137.2 Requirements

- LVV-167 - DMS-REQ-0336-V-01: Regenerating Data Products from Previous Data Releases

#### 4.137.3 Test Script

**Step 1**
Delegate to LSP

### 4.138 LVV-T160 - Verify implementation of Providing a Precovery Service (DMS-REQ-0341)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

#### 4.138.1 Test Items

LSP Portal, MOPS and Forced Photometry

#### 4.138.2 Requirements

- LVV-172 - DMS-REQ-0341-V-01: Providing a Precovery Service

#### 4.138.3 Precondition

1. DECam HiTS data could be an appropriate set for this activity.
2. Precovery pipelines for follow-on to alert processing must exist and be made available as a containerized version within the Science Platform.
3. Determine limitations over which general precovery is supported. I would suggest that precovery services be limited to current (or last two) DRP campaigns with the possible addition of including non-DRP products to encompass observations over the preceding year (does this then require means to re-generate PVIs from Alert Production in addition to DRP?)
4. Could re-use elements of LVV-T80 where quasars are used to test faint object detection.

#### 4.138.4 Test Script

**Step 1**
Run Precovery within follow-on Alert Production (i.e. daily post-processing on 30 day store).

**Step 2**
Within Science Platform, initiate request to perform precovery for a list of sources over same period (and longer). Include among the sources for precovery quasars from LVV-T80.

**Step 3**
Examine the results. Compare the results for the period where there is overlap with precovery run... and quasar photometry with those from LVV-T80 to verify user service performs as production services.

### 4.139 LVV-T161 - Verify implementation of Logging of catalog queries (DMS-REQ-0345)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.139.1 Test Items

Managed Database, LSP Web APIs, Parallel Distributed Database

#### 4.139.2 Requirements

- LVV-176 - DMS-REQ-0345-V-01: Logging of catalog queries

#### 4.139.3 Test Script

**Step 1**
Delegate to LSP

## 4.140 LVV-T162 - Verify implementation of Access to Previous Data Releases (DMS-REQ-0363)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

### 4.140.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.140.2 Requirements

- LVV-189 - DMS-REQ-0363-V-01: Access to Previous Data Releases

### 4.140.3 Precondition

Requires two (fake) releases within DAC (or PDAC) with common area/observations (preferably with some differing results but could use metadata identifying provenance).

### 4.140.4 Test Script

**Step 1**
From Science Platform initiate request for image and catalog products from one of the two release sets.

**Step 2**
From Science Platform re-issue the same request but specifying the alternate/earlier release set.

**Step 3**
Compare results and identify differences that are germaine to the relevant Data Release Sets are found.

## 4.141 LVV-T163 - Verify implementation of Data Access Services (DMS-REQ-0364)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.141.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.141.2 Requirements

- LVV-190 - DMS-REQ-0364-V-01: Data Access Services

### 4.141.3 Test Script

**Step 1**
Delegate to LSP

## 4.142 LVV-T164 - Verify implementation of Operations Subsets (DMS-REQ-0365)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.142.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

#### 4.142.2 Requirements

- LVV-191 - DMS-REQ-0365-V-01: Operations Subsets

#### 4.142.3 Test Script

**Step 1**
Delegate to LSP

### 4.143 LVV-T165 - Verify implementation of Subsets Support (DMS-REQ-0366)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.143.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

#### 4.143.2 Requirements

- LVV-192 - DMS-REQ-0366-V-01: Subsets Support

#### 4.143.3 Test Script

**Step 1**
Delegate to LSP

### 4.144 LVV-T166 - Verify implementation of Access Services Performance (DMS-REQ-0367)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.144.1 Test Items

Data Backbone, Managed Database, Compute/Storage/LAN, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

#### 4.144.2 Requirements

- LVV-193 - DMS-REQ-0367-V-01: Access Services Performance

#### 4.144.3 Test Script

**Step 1**
Delegate to LSP

### 4.145 LVV-T167 - Verify implementation of Implementation Provisions (DMS-REQ-0368)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.145.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

#### 4.145.2 Requirements

- LVV-194 - DMS-REQ-0368-V-01: Implementation Provisions

#### 4.145.3 Test Script

**Step 1**
Delegate to LSP

### 4.146 LVV-T168 - Verify implementation of Evolution (DMS-REQ-0369)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.146.1 Test Items

Data Backbone, Managed Database, Service Management and Monitoring, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

#### 4.146.2 Requirements

- LVV-195 - DMS-REQ-0369-V-01: Evolution

#### 4.146.3 Test Script

**Step 1**
Delegate to LSP

### 4.147 LVV-T169 - Verify implementation of Older Release Behavior (DMS-REQ-0370)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.147.1 Test Items

Data Backbone, Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.147.2 Requirements

- LVV-196 - DMS-REQ-0370-V-01: Older Release Behavior

### 4.147.3 Test Script

**Step 1**
Delegate to LSP

## 4.148 LVV-T170 - Verify implementation of Query Availability (DMS-REQ-0371)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.148.1 Test Items

Managed Database, LSP Portal, LSP JupyterLab, LSP Web APIs, Parallel Distributed Database

### 4.148.2 Requirements

- LVV-197 - DMS-REQ-0371-V-01: Query Availability

### 4.148.3 Test Script

**Step 1**
Delegate to LSP

## 4.149 LVV-T171 - Verify implementation of Pipeline Availability (DMS-REQ-0008)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.149.1 Test Items

Image and EFD Archiving, Prompt Processing, OCS Driven Batch, Telemetry Gateway, Alert Distribution, Alert Filtering, Batch Production, Data Backbone, Compute/Storage/LAN, Inter-Site Networks, Service Management and Monitoring

### 4.149.2 Requirements

- LVV-5 - DMS-REQ-0008-V-01: Pipeline Availability

### 4.149.3 Test Script

**Step 1**
Analyze sources of downtime after mitigation to compute estimated reliability; observe unscheduled downtime of developer, integration, and pre-production systems

## 4.150 LVV-T172 - Verify implementation of Optimization of Cost, Reliability and Availability (DMS-REQ-0161)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.150.1 Test Items

Pointing Prediction Publishing, Alert Distribution, Alert Filtering, Data Backbone, Compute/Storage/LAN, Service Management and Monitoring, LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.150.2 Requirements

- LVV-64 - DMS-REQ-0161-V-01: Optimization of Cost, Reliability and Availability in Order

### 4.150.3 Test Script

**Step 1**
Analyze resource management policy

## 4.151 LVV-T173 - Verify implementation of Pipeline Throughput (DMS-REQ-0162)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.151.1 Test Items

Image and EFD Archiving, Prompt Processing, OCS Driven Batch, Alert Distribution, Alert Filtering, Data Backbone, Compute/Storage/LAN, Service Management and Monitoring

### 4.151.2 Requirements

- LVV-65 - DMS-REQ-0162-V-01: Pipeline Throughput

### 4.151.3 Test Script

**Step 1**
Execute single-day operations rehearsal, observe data products generated in time

## 4.152 LVV-T174 - Verify implementation of Re-processing Capacity (DMS-REQ-0163)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Robert Gruendl |

#### 4.152.1 Test Items

Batch Production, Data Backbone, Batch Computing, Compute/Storage/LAN, Inter-Site Networks, Service Management and Monitoring

#### 4.152.2 Requirements

- LVV-66 - DMS-REQ-0163-V-01: Re-processing Capacity

#### 4.152.3 Test Script

**Step 1**
Analyze sizing model; execute DRP, observe scaling

### 4.153 LVV-T175 - Verify implementation of Temporary Storage for Communications Links (DMS-REQ-0164)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.153.1 Test Items

Data Backbone, Compute/Storage/LAN

#### 4.153.2 Requirements

- LVV-67 - DMS-REQ-0164-V-01: Temporary Storage for Communications Links

#### 4.153.3 Test Script

**Step 1**
Analyze sizing model and network/storage design

### 4.154 LVV-T176 - Verify implementation of Infrastructure Sizing for ”catching up” (DMS-REQ-0165)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.154.1 Test Items

Image and EFD Archiving, Data Backbone, Batch Computing, Compute/Storage/LAN, Inter-Site Networks

#### 4.154.2 Requirements

- LVV-68 - DMS-REQ-0165-V-01: Infrastructure Sizing for ”catching up”

#### 4.154.3 Test Script

**Step 1**
Execute single-day operations rehearsal including catch-up after failure, observe data products generated in time

### 4.155 LVV-T177 - Verify implementation of Incorporate Fault-Tolerance (DMS-REQ-0166)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.155.1 Test Items

Data Backbone, Containerized Application Management, Compute/Storage/LAN, Inter-Site Networks

### 4.155.2 Requirements

- LVV-69 - DMS-REQ-0166-V-01: Incorporate Fault-Tolerance

### 4.155.3 Test Script

**Step 1**
Analyze design; execute single-day operations rehearsal including failures, observe recovery without loss of data

## 4.156 LVV-T178 - Verify implementation of Incorporate Autonomics (DMS-REQ-0167)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.156.1 Test Items

Image and EFD Archiving, Prompt Processing, Observatory Operations Data, Alert Distribution, Alert Filtering, Batch Production, Data Backbone, Containerized Application Management, Compute/Storage/LAN, Inter-Site Networks

### 4.156.2 Requirements

- LVV-70 - DMS-REQ-0167-V-01: Incorporate Autonomics

### 4.156.3 Test Script

**Step 1**
Analyze design; execute single-day operations rehearsal including failures, observe automated recovery and continuation of processing

## 4.157 LVV-T179 - Verify implementation of Compute Platform Heterogeneity (DMS-REQ-0314)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.157.1 Test Items

Image and EFD Archiving, Prompt Processing, Observatory Operations Data, OCS Driven Batch, Telemetry Gateway, Alert Distribution, Alert Filtering, Batch Production, Bulk Distribution, Developer Services, Integration and Test, Data Backbone, Batch Computing, Containerized Application Management, Compute/Storage/LAN, LSP Portal, LSP JupyterLab, LSP Web APIs, QC System, Data Butler Access Client, Task Framework

### 4.157.2 Requirements

- LVV-145 - DMS-REQ-0314-V-01: Compute Platform Heterogeneity

### 4.157.3 Test Script

**Step 1**
Configure heterogeneous cluster, execute AP+DRP+LSP, observe correct functioning

## 4.158 LVV-T180 - Verify implementation of Data Management Unscheduled Downtime (DMS-REQ-0318)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.158.1 Test Items

Image and EFD Archiving, Prompt Processing, Observatory Operations Data, OCS Driven Batch, Telemetry Gateway, Alert Distribution, Alert Filtering, Batch Production, Bulk Distribution, Data Backbone, Managed Database, Batch Computing, Containerized Application Management, Compute/Storage/LAN, Inter-Site Networks, Service Management and Monitoring, LSP Portal, LSP JupyterLab, LSP Web APIs, QC System

#### 4.158.2 Requirements

- LVV-149 - DMS-REQ-0318-V-01: Data Management Unscheduled Downtime

#### 4.158.3 Test Script

**Step 1**
Analyze likely hardware failures with mitigations to compute estimated unplanned downtime

### 4.159 LVV-T181 - Verify implementation of Summit Facility Data Communications (DMS-REQ-0168)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.159.1 Test Items

Inter-Site Networks

### 4.159.2 Requirements

- LVV-71 - DMS-REQ-0168-V-01: Summit Facility Data Communications

### 4.159.3 Test Script

**Step 1**
Delegate to Networks

## 4.160 LVV-T182 - Verify implementation of Prefer Computing and Storage Down (DMS-REQ-0170)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.160.1 Test Items

Compute/Storage/LAN

### 4.160.2 Requirements

- LVV-72 - DMS-REQ-0170-V-01: Prefer Computing and Storage Down

### 4.160.3 Test Script

**Step 1**
Analyze design

## 4.161 LVV-T183 - Verify implementation DMS Communication with OCS (DMS-REQ-0315)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.161.1 Test Items

Image and EFD Archiving, Prompt Processing, Observatory Operations Data, OCS Driven Batch, Telemetry Gateway

#### 4.161.2 Requirements

- LVV-146 - DMS-REQ-0315-V-01: DMS Communication with OCS

#### 4.161.3 Test Script

**Step 1**
Delegate to IIP

### 4.162 LVV-T184 - Verify implementation of Summit to Base Network (DMS-REQ-0171)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.162.1 Test Items

Inter-Site Networks

#### 4.162.2 Requirements

- LVV-73 - DMS-REQ-0171-V-01: Summit to Base Network

#### 4.162.3 Test Script

**Step 1**
Delegate to Networks

### 4.163 LVV-T185 - Verify implementation of Summit to Base Network Availability (DMS-REQ-0172)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.163.1 Test Items

Inter-Site Networks

#### 4.163.2 Requirements

- LVV-74 - DMS-REQ-0172-V-01: Summit to Base Network Availability

#### 4.163.3 Test Script

**Step 1**
Delegate to Networks

### 4.164 LVV-T186 - Verify implementation of Summit to Base Network Reliability (DMS-REQ-0173)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.164.1 Test Items

Inter-Site Networks

### 4.164.2 Requirements

- LVV-75 - DMS-REQ-0173-V-01: Summit to Base Network Reliability

### 4.164.3 Test Script

**Step 1**
Delegate to Networks

## 4.165 LVV-T187 - Verify implementation of Summit to Base Network Secondary Link (DMS-REQ-0174)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.165.1 Test Items

Inter-Site Networks

### 4.165.2 Requirements

- LVV-76 - DMS-REQ-0174-V-01: Summit to Base Network Secondary Link

### 4.165.3 Test Script

**Step 1**
Delegate to Networks

## 4.166 LVV-T188 - Verify implementation of Summit to Base Network Ownership and Operation (DMS-REQ-0175)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.166.1 Test Items

Inter-Site Networks

### 4.166.2 Requirements

- LVV-77 - DMS-REQ-0175-V-01: Summit to Base Network Ownership and Operation

### 4.166.3 Test Script

**Step 1**
Delegate to Networks

## 4.167 LVV-T189 - Verify implementation of Base Facility Infrastructure (DMS-REQ-0176)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.167.1 Test Items

Data Backbone, Compute/Storage/LAN

### 4.167.2 Requirements

- LVV-78 - DMS-REQ-0176-V-01: Base Facility Infrastructure

#### 4.167.3 Test Script

**Step 1**
Analyze design and sizing model

## 4.168 LVV-T190 - Verify implementation of Base Facility Co-Location with Existing Facility (DMS-REQ-0178)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.168.1 Test Items

Base Facility

### 4.168.2 Requirements

- LVV-80 - DMS-REQ-0178-V-01: Base Facility Co-Location with Existing Facility

### 4.168.3 Test Script

**Step 1**
Analyze design

## 4.169 LVV-T191 - Verify implementation of Commissioning Cluster (DMS-REQ-0316)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.169.1 Test Items

Commissioning Cluster, Compute/Storage/LAN, Base Facility

### 4.169.2 Requirements

- LVV-147 - DMS-REQ-0316-V-01: Commissioning Cluster

### 4.169.3 Test Script

**Step 1**
Analyze design and budget

## 4.170 LVV-T192 - Verify implementation of Base Wireless LAN (WiFi) (DMS-REQ-0352)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.170.1 Test Items

Compute/Storage/LAN

### 4.170.2 Requirements

- LVV-183 - DMS-REQ-0352-V-01: Base Wireless LAN (WiFi)

### 4.170.3 Test Script

**Step 1**
Delegate to Networks

## 4.171 LVV-T193 - Verify implementation of Base to Archive Network (DMS-REQ-0180)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.171.1 Test Items

Inter-Site Networks

### 4.171.2 Requirements

- LVV-81 - DMS-REQ-0180-V-01: Base to Archive Network

### 4.171.3 Test Script

**Step 1**
Delegate to Networks

## 4.172 LVV-T194 - Verify implementation of Base to Archive Network Availability (DMS-REQ-0181)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.172.1 Test Items

Inter-Site Networks

### 4.172.2 Requirements

- LVV-82 - DMS-REQ-0181-V-01: Base to Archive Network Availability

#### 4.172.3 Test Script

**Step 1**
Delegate to Networks

### 4.173 LVV-T195 - Verify implementation of Base to Archive Network Reliability (DMS-REQ-0182)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.173.1 Test Items

Inter-Site Networks

#### 4.173.2 Requirements

- LVV-83 - DMS-REQ-0182-V-01: Base to Archive Network Reliability

#### 4.173.3 Test Script

**Step 1**
Delegate to Networks

### 4.174 LVV-T196 - Verify implementation of Base to Archive Network Secondary Link (DMS-REQ-0183)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.174.1 Test Items

Inter-Site Networks

### 4.174.2 Requirements

- LVV-84 - DMS-REQ-0183-V-01: Base to Archive Network Secondary Link

### 4.174.3 Test Script

**Step 1**
Delegate to Networks

## 4.175 LVV-T197 - Verify implementation of Archive Center (DMS-REQ-0185)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.175.1 Test Items

Bulk Distribution, Data Backbone, Service Management and Monitoring, NCSA Facility

### 4.175.2 Requirements

- LVV-85 - DMS-REQ-0185-V-01: Archive Center

### 4.175.3 Test Script

**Step 1**
Analyze design and sizing model

## 4.176 LVV-T198 - Verify implementation of Archive Center Disaster Recovery (DMS-REQ-0186)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.176.1 Test Items

Bulk Distribution, Data Backbone, Compute/Storage/LAN, Service Management and Monitoring

### 4.176.2 Requirements

- LVV-86 - DMS-REQ-0186-V-01: Archive Center Disaster Recovery

### 4.176.3 Test Script

**Step 1**
Analyze design; simulate storage failure, observe restore from disaster recovery

## 4.177 LVV-T199 - Verify implementation of Archive Center Co-Location with Existing Facility (DMS-REQ-0187)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.177.1 Test Items

NCSA Facility

### 4.177.2 Requirements

- LVV-87 - DMS-REQ-0187-V-01: Archive Center Co-Location with Existing Facility

### 4.177.3 Test Script

**Step 1**
Analyze design

## 4.178 LVV-T200 - Verify implementation of Archive to Data Access Center Network (DMS-REQ-0188)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.178.1 Test Items

Inter-Site Networks

### 4.178.2 Requirements

- LVV-88 - DMS-REQ-0188-V-01: Archive to Data Access Center Network

### 4.178.3 Test Script

**Step 1**
Delegate to Networks

## 4.179 LVV-T201 - Verify implementation of Archive to Data Access Center Network Availability (DMS-REQ-0189)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.179.1 Test Items

Inter-Site Networks

#### 4.179.2 Requirements

- LVV-89 - DMS-REQ-0189-V-01: Archive to Data Access Center Network Availability

#### 4.179.3 Test Script

**Step 1**
Delegate to Networks

### 4.180 LVV-T202 - Verify implementation of Archive to Data Access Center Network Reliability (DMS-REQ-0190)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.180.1 Test Items

Inter-Site Networks

#### 4.180.2 Requirements

- LVV-90 - DMS-REQ-0190-V-01: Archive to Data Access Center Network Reliability

### 4.180.3 Test Script

**Step 1**
Delegate to Networks

## 4.181 LVV-T203 - Verify implementation of Archive to Data Access Center Network Secondary Link (DMS-REQ-0191)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.181.1 Test Items

Inter-Site Networks

### 4.181.2 Requirements

- LVV-91 - DMS-REQ-0191-V-01: Archive to Data Access Center Network Secondary Link

### 4.181.3 Test Script

**Step 1**
Take primary network link down

**Step 2**
Observe operations support over secondary link

**Step 3**
Bring primary network link back up

**Step 4**
Observe catch-up capability over secondary link

## 4.182 LVV-T204 - Verify implementation of Access to catalogs for external Level 3 processing (DMS-REQ-0122)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.182.1 Test Items

Execute bulk distribution of DRP catalogs, observe correct transfer and use of maintenance/-validation tools

### 4.182.2 Requirements

- LVV-50 - DMS-REQ-0122-V-01: Access to catalogs for external Level 3 processing

### 4.182.3 Test Script

**Step 1**
Execute bulk distribution of DRP catalogs

**Step 2**
Observe correct transfer and use of maintenance/validation tools

## 4.183 LVV-T205 - Verify implementation of Access to input catalogs for DAC-based Level 3 processing (DMS-REQ-0123)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

#### 4.183.1 Test Items

Bulk Distribution, Data Backbone, LSP Portal, LSP JupyterLab, LSP Web APIs

#### 4.183.2 Requirements

- LVV-51 - DMS-REQ-0123-V-01: Access to input catalogs for DAC-based Level 3 processing

#### 4.183.3 Test Script

**Step 1**
Load Prompt and DR catalogs into PDAC, observe access via LSP

### 4.184 LVV-T206 - Verify implementation of Federation with external catalogs (DMS-REQ-0124)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Colin Slater |

#### 4.184.1 Test Items

Data Backbone, LSP Portal, LSP JupyterLab, LSP Web APIs

#### 4.184.2 Requirements

- LVV-52 - DMS-REQ-0124-V-01: Federation with external catalogs

#### 4.184.3 Test Script

**Step 1**
Load external catalog into PDAC (using VO if possible), observe federation with other catalogs via LSP

## 4.185 LVV-T207 - Verify implementation of Access to images for external Level 3 processing (DMS-REQ-0126)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.185.1 Test Items

Execute bulk distribution of DRP images, observe correct transfer and use of maintenance/-validation tools

### 4.185.2 Requirements

- LVV-54 - DMS-REQ-0126-V-01: Access to images for external Level 3 processing

### 4.185.3 Test Script

**Step 1**
Execute bulk distribution of DRP images

**Step 2**
Observe correct transfer and use of maintenance/validation tools

## 4.186 LVV-T208 - Verify implementation of Access to input images for DAC-based Level 3 processing (DMS-REQ-0127)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.186.1 Test Items

Load Prompt and DR images into PDAC, observe access via LSP

### 4.186.2 Requirements

- LVV-55 - DMS-REQ-0127-V-01: Access to input images for DAC-based Level 3 processing

### 4.186.3 Test Script

**Step 1**
Load Prompt and DR images into PDAC

**Step 2**
Observe access via LSP

## 4.187 LVV-T209 - Verify implementation of Data Access Centers (DMS-REQ-0193)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Analysis | False | Kian-Tat Lim |

### 4.187.1 Test Items

Analyze design

### 4.187.2 Requirements

- LVV-92 - DMS-REQ-0193-V-01: Data Access Centers

### 4.187.3 Test Script

**Step 1**
Analyze design

## 4.188 LVV-T210 - Verify implementation of Data Access Center Simultaneous Connections (DMS-REQ-0194)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Kian-Tat Lim |

### 4.188.1 Test Items

Simulate data access to PDAC, observe scaling

### 4.188.2 Requirements

- LVV-93 - DMS-REQ-0194-V-01: Data Access Center Simultaneous Connections

### 4.188.3 Test Script

**Step 1**
Simulate data access to PDAC

**Step 2**
Observe scaling

## 4.189 LVV-T211 - Verify implementation of Data Access Center Geographical Distribution (DMS-REQ-0196)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Analysis | False | Kian-Tat Lim |

### 4.189.1 Test Items

Analyze design

### 4.189.2 Requirements

- LVV-94 - DMS-REQ-0196-V-01: Data Access Center Geographical Distribution

### 4.189.3 Test Script

**Step 1**
Analyze design

## 4.190 LVV-T212 - Verify implementation of No Limit on Data Access Centers (DMS-REQ-0197)

| Version | Status | Priority | Verification Type | Critical Event | Owner |
|---|---|---|---|---|---|
| 1 | Draft | Normal | Test | False | Leanne Guy |

### 4.190.1 Test Items

Data Backbone, LSP Portal, LSP JupyterLab, LSP Web APIs

### 4.190.2 Requirements

- LVV-95 - DMS-REQ-0197-V-01: No Limit on Data Access Centers

### 4.190.3 Test Script

**Step 1**
Analyze design; instantiate and load simulated DAC, observe correct functioning

# 5 Requirements Traceability

| Requirements | Test Cases |
|---|---|
| LVV-3 - DMS-REQ-0002-V-01: Transient Alert Distribution | LVV-T101 |
| LVV-4 - DMS-REQ-0004-V-01: Nightly Data Accessible Within 24 hrs | LVV-T35 |
| LVV-5 - DMS-REQ-0008-V-01: Pipeline Availability | LVV-T171 |
| LVV-6 - DMS-REQ-0009-V-01: Simulated Data | LVV-T125 |
| LVV-7 - DMS-REQ-0010-V-01: Difference Exposures | LVV-T36 |
| LVV-8 - DMS-REQ-0018-V-01: Raw Science Image Data Acquisition | LVV-T29 |
| LVV-9 - DMS-REQ-0020-V-01: Wavefront Sensor Data Acquisition | LVV-T30 |
| LVV-10 - DMS-REQ-0022-V-01: Crosstalk Corrected Science Image Data Acquisition | LVV-T31 |
| LVV-11 - DMS-REQ-0024-V-01: Raw Image Assembly | LVV-T32 |
| LVV-12 - DMS-REQ-0029-V-01: Generate Photometric Zeropoint for Visit Image | LVV-T39 |
| LVV-13 - DMS-REQ-0030-V-01: Generate WCS for Visit Images | LVV-T40 |
| LVV-14 - DMS-REQ-0032-V-01: Image Differencing | LVV-T126 |
| LVV-15 - DMS-REQ-0033-V-01: Provide Source Detection Software | LVV-T127 |
| LVV-16 - DMS-REQ-0034-V-01: Associate Sources to Objects | LVV-T61 |
| LVV-17 - DMS-REQ-0042-V-01: Provide Astrometric Model | LVV-T128 |
| LVV-18 - DMS-REQ-0043-V-01: Provide Calibrated Photometry | LVV-T129 |
| LVV-19 - DMS-REQ-0046-V-01: Provide Photometric Redshifts of Galaxies | LVV-T68 |
| LVV-20 - DMS-REQ-0047-V-01: Provide PSF for Coadded Images | LVV-T62 |
| LVV-21 - DMS-REQ-0052-V-01: Enable a Range of Shape Measurement Approaches | LVV-T130 |
| LVV-22 - DMS-REQ-0059-V-01: Bad Pixel Map | LVV-T83 |
| LVV-23 - DMS-REQ-0060-V-01: Bias Residual Image | LVV-T84 |
| LVV-24 - DMS-REQ-0061-V-01: Crosstalk Correction Matrix | LVV-T85 |
| LVV-25 - DMS-REQ-0062-V-01: Illumination Correction Frame | LVV-T86 |
| LVV-26 - DMS-REQ-0063-V-01: Monochromatic Flatfield Data Cube | LVV-T87 |
| LVV-27 - DMS-REQ-0065-V-01: Provide Image Access Services | LVV-T134 |
| LVV-28 - DMS-REQ-0068-V-01: Raw Science Image Metadata | LVV-T33 |
| LVV-29 - DMS-REQ-0069-V-01: Processed Visit Images | LVV-T38 |
| LVV-30 - DMS-REQ-0070-V-01: Generate PSF for Visit Images | LVV-T41 |
| LVV-31 - DMS-REQ-0072-V-01: Processed Visit Image Content | LVV-T42 |
| LVV-32 - DMS-REQ-0074-V-01: Difference Exposure Attributes | LVV-T37 |

| Requirements | Test Cases |
|---|---|
| LVV-33 - DMS-REQ-0075-V-01: Catalog Queries | LVV-T149 |
| LVV-34 - DMS-REQ-0077-V-01: Maintain Archive Publicly Accessible | LVV-T150 |
| LVV-35 - DMS-REQ-0078-V-01: Catalog Export Formats | LVV-T151 |
| LVV-36 - DMS-REQ-0089-V-01: Solar System Objects Available Within Specified Time | LVV-T102 |
| LVV-37 - DMS-REQ-0094-V-01: Keep Historical Alert Archive | LVV-T152 |
| LVV-38 - DMS-REQ-0096-V-01: Generate Data Quality Report Within Specified Time | LVV-T103 |
| LVV-39 - DMS-REQ-0097-V-01: Level 1 Data Quality Report Definition | LVV-T45 |
| LVV-40 - DMS-REQ-0098-V-01: Generate DMS Performance Report Within Specified Time | LVV-T104 |
| LVV-41 - DMS-REQ-0099-V-01: Level 1 Performance Report Definition | LVV-T46 |
| LVV-42 - DMS-REQ-0100-V-01: Generate Calibration Report Within Specified Time | LVV-T105 |
| LVV-43 - DMS-REQ-0101-V-01: Level 1 Calibration Report Definition | LVV-T47 |
| LVV-44 - DMS-REQ-0102-V-01: Provide Engineering & Facility Database Archive | LVV-T153 |
| LVV-45 - DMS-REQ-0103-V-01: Produce Images for EPO | LVV-T63 |
| LVV-46 - DMS-REQ-0106-V-01: Coadded Image Provenance | LVV-T64 |
| LVV-47 - DMS-REQ-0119-V-01: DAC resource allocation for Level 3 processing | LVV-T117 |
| LVV-48 - DMS-REQ-0120-V-01: Level 3 Data Product Self Consistency | LVV-T118 |
| LVV-49 - DMS-REQ-0121-V-01: Provenance for Level 3 processing at DACs | LVV-T119 |
| LVV-50 - DMS-REQ-0122-V-01: Access to catalogs for external Level 3 processing | LVV-T204 |
| LVV-51 - DMS-REQ-0123-V-01: Access to input catalogs for DAC-based Level 3 processing | LVV-T205 |
| LVV-52 - DMS-REQ-0124-V-01: Federation with external catalogs | LVV-T206 |
| LVV-53 - DMS-REQ-0125-V-01: Software framework for Level 3 catalog processing | LVV-T120 |
| LVV-54 - DMS-REQ-0126-V-01: Access to images for external Level 3 processing | LVV-T207 |
| LVV-55 - DMS-REQ-0127-V-01: Access to input images for DAC-based Level 3 processing | LVV-T208 |

| Requirements | Test Cases |
|---|---|
| LVV-56 - DMS-REQ-0128-V-01: Software framework for Level 3 image processing | LVV-T121 |
| LVV-57 - DMS-REQ-0130-V-01: Calibration Data Products | LVV-T88 |
| LVV-58 - DMS-REQ-0131-V-01: Calibration Images Available Within Specified Time | LVV-T106 |
| LVV-59 - DMS-REQ-0132-V-01: Calibration Image Provenance | LVV-T89 |
| LVV-60 - DMS-REQ-0155-V-01: Provide Data Access Services | LVV-T135 |
| LVV-61 - DMS-REQ-0156-V-01: Provide Pipeline Execution Services | LVV-T139 |
| LVV-62 - DMS-REQ-0158-V-01: Provide Pipeline Construction Services | LVV-T143 |
| LVV-63 - DMS-REQ-0160-V-01: Provide User Interface Services | LVV-T131 |
| LVV-64 - DMS-REQ-0161-V-01: Optimization of Cost, Reliability and Availability in Order | LVV-T172 |
| LVV-65 - DMS-REQ-0162-V-01: Pipeline Throughput | LVV-T173 |
| LVV-66 - DMS-REQ-0163-V-01: Re-processing Capacity | LVV-T174 |
| LVV-67 - DMS-REQ-0164-V-01: Temporary Storage for Communications Links | LVV-T175 |
| LVV-68 - DMS-REQ-0165-V-01: Infrastructure Sizing for ”catching up” | LVV-T176 |
| LVV-69 - DMS-REQ-0166-V-01: Incorporate Fault-Tolerance | LVV-T177 |
| LVV-70 - DMS-REQ-0167-V-01: Incorporate Autonomics | LVV-T178 |
| LVV-71 - DMS-REQ-0168-V-01: Summit Facility Data Communications | LVV-T181 |
| LVV-72 - DMS-REQ-0170-V-01: Prefer Computing and Storage Down | LVV-T182 |
| LVV-73 - DMS-REQ-0171-V-01: Summit to Base Network | LVV-T184 |
| LVV-74 - DMS-REQ-0172-V-01: Summit to Base Network Availability | LVV-T185 |
| LVV-75 - DMS-REQ-0173-V-01: Summit to Base Network Reliability | LVV-T186 |
| LVV-76 - DMS-REQ-0174-V-01: Summit to Base Network Secondary Link | LVV-T187 |
| LVV-77 - DMS-REQ-0175-V-01: Summit to Base Network Ownership and Operation | LVV-T188 |
| LVV-78 - DMS-REQ-0176-V-01: Base Facility Infrastructure | LVV-T189 |
| LVV-80 - DMS-REQ-0178-V-01: Base Facility Co-Location with Existing Facility | LVV-T190 |
| LVV-81 - DMS-REQ-0180-V-01: Base to Archive Network | LVV-T193 |
| LVV-82 - DMS-REQ-0181-V-01: Base to Archive Network Availability | LVV-T194 |
| LVV-83 - DMS-REQ-0182-V-01: Base to Archive Network Reliability | LVV-T195 |
| LVV-84 - DMS-REQ-0183-V-01: Base to Archive Network Secondary Link | LVV-T196 |

| Requirements | Test Cases |
|---|---|
| LVV-85 - DMS-REQ-0185-V-01: Archive Center | LVV-T197 |
| LVV-86 - DMS-REQ-0186-V-01: Archive Center Disaster Recovery | LVV-T198 |
| LVV-87 - DMS-REQ-0187-V-01: Archive Center Co-Location with Existing Facility | LVV-T199 |
| LVV-88 - DMS-REQ-0188-V-01: Archive to Data Access Center Network | LVV-T200 |
| LVV-89 - DMS-REQ-0189-V-01: Archive to Data Access Center Network Availability | LVV-T201 |
| LVV-90 - DMS-REQ-0190-V-01: Archive to Data Access Center Network Reliability | LVV-T202 |
| LVV-91 - DMS-REQ-0191-V-01: Archive to Data Access Center Network Secondary Link | LVV-T203 |
| LVV-92 - DMS-REQ-0193-V-01: Data Access Centers | LVV-T209 |
| LVV-93 - DMS-REQ-0194-V-01: Data Access Center Simultaneous Connections | LVV-T210 |
| LVV-94 - DMS-REQ-0196-V-01: Data Access Center Geographical Distribution | LVV-T211 |
| LVV-95 - DMS-REQ-0197-V-01: No Limit on Data Access Centers | LVV-T212 |
| LVV-96 - DMS-REQ-0265-V-01: Guider Calibration Data Acquisition | LVV-T34 |
| LVV-97 - DMS-REQ-0266-V-01: Exposure Catalog | LVV-T48 |
| LVV-98 - DMS-REQ-0267-V-01: Source Catalog | LVV-T65 |
| LVV-99 - DMS-REQ-0268-V-01: Forced-Source Catalog | LVV-T66 |
| LVV-100 - DMS-REQ-0269-V-01: DIASource Catalog | LVV-T49 |
| LVV-101 - DMS-REQ-0270-V-01: Faint DIASource Measurements | LVV-T50 |
| LVV-102 - DMS-REQ-0271-V-01: DIAObject Catalog | LVV-T51 |
| LVV-103 - DMS-REQ-0272-V-01: DIAObject Attributes | LVV-T52 |
| LVV-104 - DMS-REQ-0273-V-01: SSObject Catalog | LVV-T53 |
| LVV-105 - DMS-REQ-0274-V-01: Alert Content | LVV-T54 |
| LVV-106 - DMS-REQ-0275-V-01: Object Catalog | LVV-T67 |
| LVV-107 - DMS-REQ-0276-V-01: Object Characterization | LVV-T69 |
| LVV-108 - DMS-REQ-0277-V-01: Coadd Source Catalog | LVV-T70 |
| LVV-109 - DMS-REQ-0278-V-01: Coadd Image Method Constraints | LVV-T72 |
| LVV-110 - DMS-REQ-0279-V-01: Deep Detection Coadds | LVV-T73 |
| LVV-111 - DMS-REQ-0280-V-01: Template Coadds | LVV-T74 |
| LVV-112 - DMS-REQ-0281-V-01: Multi-band Coadds | LVV-T75 |

| Requirements | Test Cases |
|---|---|
| LVV-113 - DMS-REQ-0282-V-01: Dark Current Correction Frame | LVV-T90 |
| LVV-114 - DMS-REQ-0283-V-01: Fringe Correction Frame | LVV-T91 |
| LVV-115 - DMS-REQ-0284-V-01: Level-1 Production Completeness | LVV-T107 |
| LVV-116 - DMS-REQ-0285-V-01: Level 1 Source Association | LVV-T108 |
| LVV-117 - DMS-REQ-0286-V-01: SSObject Precovery | LVV-T109 |
| LVV-118 - DMS-REQ-0287-V-01: DIASource Precovery | LVV-T110 |
| LVV-119 - DMS-REQ-0288-V-01: Use of External Orbit Catalogs | LVV-T111 |
| LVV-120 - DMS-REQ-0289-V-01: Calibration Production Processing | LVV-T115 |
| LVV-121 - DMS-REQ-0290-V-01: Level 3 Data Import | LVV-T122 |
| LVV-122 - DMS-REQ-0291-V-01: Query Repeatability | LVV-T96 |
| LVV-123 - DMS-REQ-0292-V-01: Uniqueness of IDs Across Data Releases | LVV-T97 |
| LVV-124 - DMS-REQ-0293-V-01: Selection of Datasets | LVV-T98 |
| LVV-125 - DMS-REQ-0294-V-01: Processing of Datasets | LVV-T99 |
| LVV-126 - DMS-REQ-0295-V-01: Transparent Data Access | LVV-T100 |
| LVV-127 - DMS-REQ-0296-V-01: Pre-cursor, and Real Data | LVV-T132 |
| LVV-128 - DMS-REQ-0297-V-01: DMS Initialization Component | LVV-T146 |
| LVV-129 - DMS-REQ-0298-V-01: Data Product and Raw Data Access | LVV-T136 |
| LVV-130 - DMS-REQ-0299-V-01: Data Product Ingest | LVV-T137 |
| LVV-131 - DMS-REQ-0300-V-01: Bulk Download Service | LVV-T138 |
| LVV-132 - DMS-REQ-0301-V-01: Control of Level-1 Production | LVV-T147 |
| LVV-133 - DMS-REQ-0302-V-01: Production Orchestration | LVV-T140 |
| LVV-134 - DMS-REQ-0303-V-01: Production Monitoring | LVV-T141 |
| LVV-135 - DMS-REQ-0304-V-01: Production Fault Tolerance | LVV-T142 |
| LVV-136 - DMS-REQ-0305-V-01: Task Specification | LVV-T144 |
| LVV-137 - DMS-REQ-0306-V-01: Task Configuration | LVV-T145 |
| LVV-138 - DMS-REQ-0307-V-01: Unique Processing Coverage | LVV-T148 |
| LVV-139 - DMS-REQ-0308-V-01: Software Architecture to Enable Community Re-Use | LVV-T124 |
| LVV-140 - DMS-REQ-0309-V-01: Raw Data Archiving Reliability | LVV-T154 |
| LVV-141 - DMS-REQ-0310-V-01: Un-Archived Data Product Cache | LVV-T155 |
| LVV-142 - DMS-REQ-0311-V-01: Regenerate Un-archived Data Products | LVV-T156 |
| LVV-143 - DMS-REQ-0312-V-01: Level 1 Data Product Access | LVV-T157 |
| LVV-144 - DMS-REQ-0313-V-01: Level 1 & 2 Catalog Access | LVV-T158 |
| LVV-145 - DMS-REQ-0314-V-01: Compute Platform Heterogeneity | LVV-T179 |

| Requirements | Test Cases |
|---|---|
| LVV-146 - DMS-REQ-0315-V-01: DMS Communication with OCS | LVV-T183 |
| LVV-147 - DMS-REQ-0316-V-01: Commissioning Cluster | LVV-T191 |
| LVV-148 - DMS-REQ-0317-V-01: DIAForcedSource Catalog | LVV-T55 |
| LVV-149 - DMS-REQ-0318-V-01: Data Management Unscheduled Downtime | LVV-T180 |
| LVV-150 - DMS-REQ-0319-V-01: Characterizing Variability | LVV-T56 |
| LVV-151 - DMS-REQ-0320-V-01: Processing of Data From Special Programs | LVV-T92 |
| LVV-152 - DMS-REQ-0321-V-01: Level 1 Processing of Special Programs Data | LVV-T93 |
| LVV-153 - DMS-REQ-0322-V-01: Special Programs Database | LVV-T94 |
| LVV-154 - DMS-REQ-0323-V-01: Calculating SSObject Parameters | LVV-T57 |
| LVV-155 - DMS-REQ-0324-V-01: Matching DIASources to Objects | LVV-T58 |
| LVV-156 - DMS-REQ-0325-V-01: Regenerating L1 Data Products During Data Release Processing | LVV-T59 |
| LVV-157 - DMS-REQ-0326-V-01: Storing Approximations of Per-pixel Metadata | LVV-T23 |
| LVV-158 - DMS-REQ-0327-V-01: Background Model Calculation | LVV-T43 |
| LVV-159 - DMS-REQ-0328-V-01: Documenting Image Characterization | LVV-T44 |
| LVV-160 - DMS-REQ-0329-V-01: All-Sky Visualization of Data Releases | LVV-T76 |
| LVV-161 - DMS-REQ-0330-V-01: Best Seeing Coadds | LVV-T77 |
| LVV-162 - DMS-REQ-0331-V-01: Computing Derived Quantities | LVV-T24 |
| LVV-163 - DMS-REQ-0332-V-01: Denormalizing Database Tables | LVV-T25 |
| LVV-164 - DMS-REQ-0333-V-01: Maximum Likelihood Values and Covariances | LVV-T26 |
| LVV-165 - DMS-REQ-0334-V-01: Persisting Data Products | LVV-T78 |
| LVV-166 - DMS-REQ-0335-V-01: PSF-Matched Coadds | LVV-T79 |
| LVV-167 - DMS-REQ-0336-V-01: Regenerating Data Products from Previous Data Releases | LVV-T159 |
| LVV-168 - DMS-REQ-0337-V-01: Detecting faint variable objects | LVV-T80 |
| LVV-169 - DMS-REQ-0338-V-01: Targeted Coadds | LVV-T81 |
| LVV-170 - DMS-REQ-0339-V-01: Tracking Characterization Changes Between Data Releases | LVV-T82 |
| LVV-171 - DMS-REQ-0340-V-01: Access Controls of Level 3 Data Products | LVV-T123 |
| LVV-172 - DMS-REQ-0341-V-01: Providing a Precovery Service | LVV-T160 |

| Requirements | Test Cases |
|---|---|
| LVV-173 - DMS-REQ-0342-V-01: Alert Filtering Service | LVV-T112 |
| LVV-174 - DMS-REQ-0343-V-01: Performance Requirements for LSST Alert Filtering Service | LVV-T113 |
| LVV-175 - DMS-REQ-0344-V-01: Constraints on Level 1 Special Program Products Generation | LVV-T95 |
| LVV-176 - DMS-REQ-0345-V-01: Logging of catalog queries | LVV-T161 |
| LVV-177 - DMS-REQ-0346-V-01: Data Availability | LVV-T27 |
| LVV-178 - DMS-REQ-0347-V-01: Measurements in catalogs | LVV-T28 |
| LVV-179 - DMS-REQ-0348-V-01: Pre-defined alert filters | LVV-T114 |
| LVV-180 - DMS-REQ-0349-V-01: Detecting extended low surface brightness objects | LVV-T71 |
| LVV-181 - DMS-REQ-0350-V-01: Associating Objects across data releases | LVV-T116 |
| LVV-182 - DMS-REQ-0351-V-01: Provide Beam Projector Coordinate Calculation Software | LVV-T133 |
| LVV-183 - DMS-REQ-0352-V-01: Base Wireless LAN (WiFi) | LVV-T192 |
| LVV-184 - DMS-REQ-0353-V-01: Publishing predicted visit schedule | LVV-T60 |
| LVV-189 - DMS-REQ-0363-V-01: Access to Previous Data Releases | LVV-T162 |
| LVV-190 - DMS-REQ-0364-V-01: Data Access Services | LVV-T163 |
| LVV-191 - DMS-REQ-0365-V-01: Operations Subsets | LVV-T164 |
| LVV-192 - DMS-REQ-0366-V-01: Subsets Support | LVV-T165 |
| LVV-193 - DMS-REQ-0367-V-01: Access Services Performance | LVV-T166 |
| LVV-194 - DMS-REQ-0368-V-01: Implementation Provisions | LVV-T167 |
| LVV-195 - DMS-REQ-0369-V-01: Evolution | LVV-T168 |
| LVV-196 - DMS-REQ-0370-V-01: Older Release Behavior | LVV-T169 |
| LVV-197 - DMS-REQ-0371-V-01: Query Availability | LVV-T170 |