Rubin Observatory

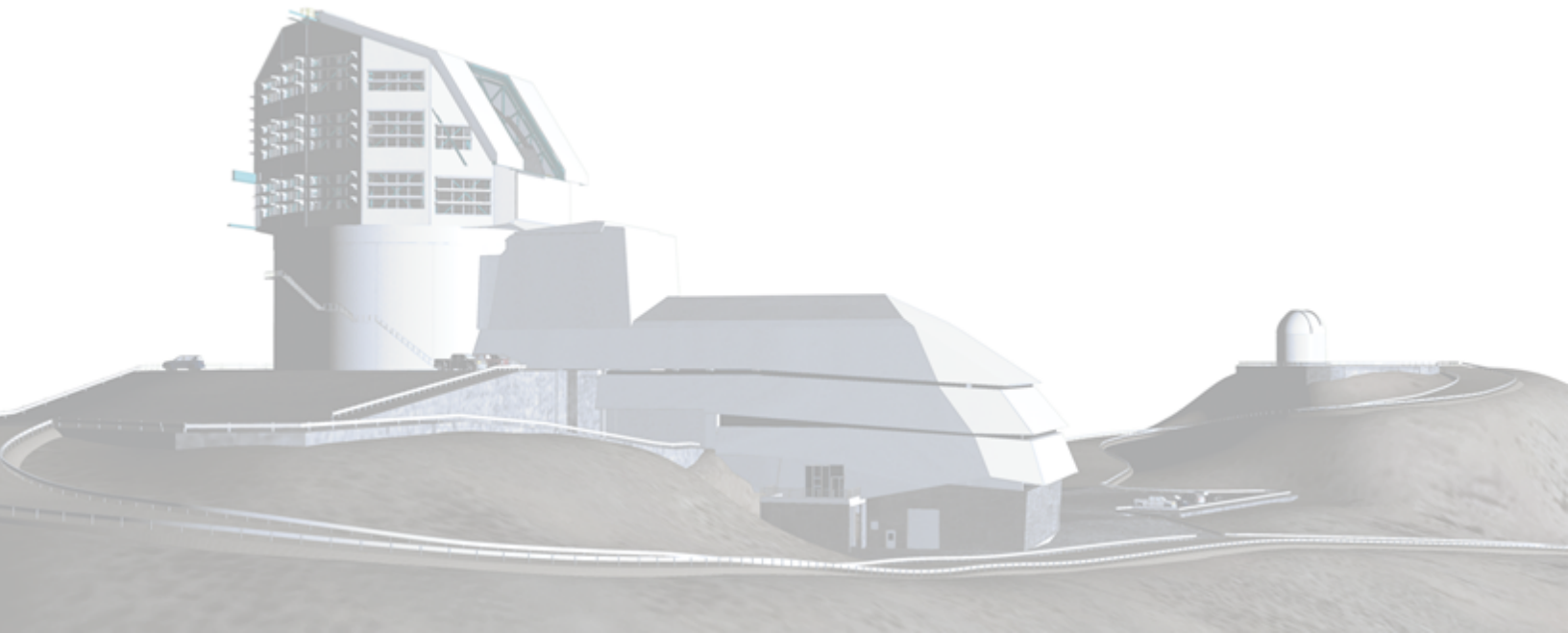
Vera C. Rubin Observatory
Data Management

LVV-P71: Science Pipelines Release 20.0.0 Acceptance Test Campaign Test Plan and Report

Jeffrey Carlin

DMTR-261

Latest Revision: 2020-08-05



Abstract

This is the test plan and report for LVV-P71 (Science Pipelines Release 20.0.0 Acceptance Test Campaign), an LSST milestone pertaining to the Data Management Subsystem.

Rubin Observatory

Change Record

Version	Date	Description	Owner name
	2020-06-24	First draft	Jeff Carlin
1.0	2020-08-05	Test plan LVV-P71 approved, test activity ready to start. DM-25646	Jeff Carlin

Document curator: Jeff Carlin
Document source location: `https://github.com/lsst-dm/DMTR-261`
Version from source repository: 779a44b

Contents

Rubin Observatory

LVV-P71: Science Pipelines Release 20.0.0 Acceptance Test Campaign Test Plan and Report

1 Introduction

1.1 Objectives

This Acceptance Test campaign aims to verify a subset of DMSR (LSE-61) requirements related to the LSST Science Pipelines. It will be executed in conjunction with the release of Science Pipelines Version 20.0.0, but the pipeline release is not contingent upon this test campaign. This Test Plan aims to demonstrate that the included requirements have been met by Version 20.0.0 of the Pipelines, and to thus fully verify their completion and readiness for LSST Operations.

1.2 System Overview

The tests to be executed are intended to verify that the DM system satisfies a subset of the requirements outlined in the Data Management System Requirements (DMSR; LSE-61). This subset of requirements is related to pipeline algorithms, and was selected for this campaign to coincide with the release of a new version of the LSST Science Pipelines. Additional DMSR requirements will be verified in later Acceptance Test Campaigns.

Applicable Documents:

LSE-61 Data Management System Requirements
LDM-503 Data Management Test Plan
LDM-639 LSST Data Management Acceptance Test Specification (issue 2.1)

The tests will be performed using the HSC-RC2 dataset (as defined in DM-11345). When possible, we will start our tests with the data products resulting from processing HSC-RC2 with the v20_0_0_rc1 pipelines release candidate (DM-24478) that was used to create v20 of

the Science Pipelines.

1.3 Document Overview

This document was generated from Jira, obtaining the relevant information from the LVV-P71 Jira Test Plan and related Test Cycles (LVV-C153).

Section 1 provides an overview of the test campaign, the system under test (Acceptance), the applicable documentation, and explains how this document is organized. Section 2 provides additional information about the test plan, like for example the configuration used for this test or related documentation. Section 3 describes the necessary roles and lists the individuals assigned to them.

Section 4 provides a summary of the test results, including an overview in Table 2, an overall assessment statement and suggestions for possible improvements. Section 5 provides detailed results for each step in each test case.

The current status of test plan LVV-P71 in Jira is **Approved** .

1.4 References

[1] **[LSE-61]**, Dubois-Felsmann, G., Jenness, T., 2018, *LSST Data Management Subsystem Requirements*, LSE-61, URL `https://ls.st/LSE-61`

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

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

2 Test Plan Details

2.1 Data Collection

Observing is not required for this test campaign.

2.2 Verification Environment

The “lsst-lsp-stable” instance of the LSST Science Platform (LSP), hosted at the LDF, and the “lsst-dev” development cluster at NCSA. In particular, we will use Release 20.0.0 of the Pipelines.

2.3 Entry Criteria

Release and availability of Science Pipelines version 20.

2.4 Related Documentation

The documentation related to this test campaign should be provided in the following DocuShare Collection (as per Verification Artifacts in Jira test plan LVV-P71).

- DocuShare Collection Not Specified

2.5 PMCS Activity

Primavera milestones related to the test campaign.

- None

3 Personnel

The personnel involved in the test campaign is shown in the following table.

	T. Plan LVV-P71 owner:	**Jeffrey Carlin**	
	T. Cycle LVV-C153 owner:	**Jeffrey Carlin**	
Test Cases	**Assigned to**	**Executed by**	**Additional Test Personnel**
LVV-T28	Colin Slater		
LVV-T133	Robert Lupton		
LVV-T1087	Fritz Mueller		
LVV-T1086	Fritz Mueller		
LVV-T1085	Fritz Mueller		
LVV-T1232	Colin Slater		
LVV-T40	Jim Bosch		
LVV-T1759	Jeffrey Carlin		
LVV-T1758	Jeffrey Carlin		
LVV-T1756	Jeffrey Carlin		
LVV-T1757	Jeffrey Carlin		
LVV-T125	Robert Lupton		
LVV-T36	Eric Bellm		
LVV-T126	Eric Bellm		
LVV-T39	Jim Bosch		
LVV-T46	Eric Bellm		
LVV-T38	Eric Bellm		
LVV-T42	Jim Bosch		
LVV-T149	Colin Slater		
LVV-T151	Colin Slater		
LVV-T45	Eric Bellm		
LVV-T146	Robert Gruendl		
LVV-T144	Kian-Tat Lim		
LVV-T145	Robert Lupton		
LVV-T1264	Robert Gruendl		

4 Test Campaign Overview

4.1 Summary

T. Plan LVV-P71:	**Science Pipelines Release 20.0.0 Acceptance Test Campaign**			Approved
T. Cycle LVV-C153:	**Pipelines v20 Release DM Acceptance Test Campaign**			Not Executed
Test Cases	**Ver.**	**Status**	**Comment**	**Issues**
LVV-T28	1	Not Executed		
LVV-T133	1	Not Executed		
LVV-T1087	1	Not Executed		
LVV-T1086	1	Not Executed		
LVV-T1085	1	Not Executed		
LVV-T1232	1	Not Executed		
LVV-T40	1	Not Executed		
LVV-T1759	1	Not Executed		
LVV-T1758	1	Not Executed		
LVV-T1756	1	Not Executed		
LVV-T1757	1	Not Executed		
LVV-T125	1	Not Executed		
LVV-T36	1	Not Executed		
LVV-T126	1	Not Executed		
LVV-T39	1	Not Executed		
LVV-T46	1	Not Executed		
LVV-T38	1	Not Executed		
LVV-T42	1	Not Executed		
LVV-T149	1	Not Executed		
LVV-T151	1	Not Executed		
LVV-T45	1	Not Executed		
LVV-T146	1	Not Executed		
LVV-T144	1	Not Executed		
LVV-T145	1	Not Executed		
LVV-T1264	1	Not Executed		

Table 2: Test Campaign Summary

4.2 Overall Assessment

Not yet available.

4.3 Recommended Improvements

Not yet available.

5 Detailed Test Results

5.1 Test Cycle LVV-C153

Open test cycle *Pipelines v20 Release DM Acceptance Test Campaign* in Jira.

Test Cycle name: Pipelines v20 Release DM Acceptance Test Campaign
Status: Not Executed

This test cycle verifies a subset of DMSR (LSE-61) requirements related to the LSST Science Pipelines, in order to verify their completion and readiness for LSST Operations (i.e., that the requirements laid out in LSE-61 have been met by the DM Systems).

5.1.1 Software Version/Baseline

All tests will be performed with LSST Science Pipelines release version 20.0.0, including its algorithms and resulting science data products.

5.1.2 Configuration

Not provided.

5.1.3 Test Cases in LVV-C153 Test Cycle

5.1.3.1 LVV-T28 - Verify implementation of Measurements in catalogs

Version **1**. Open *LVV-T28* test case in Jira.

Verify that source measurements in catalogs are in flux units.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details

1	Description
	Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following:
	Example Code
	`import lsst.daf.persistence as dafPersist`
	`butler = dafPersist.Butler(inputs='DATA/path')`
	Expected Result
	Butler repo available for reading.
	Actual Result
	Status: **Not Executed**

2	Description Identify and read appropriate processed precursor datasets with the Butler, including one containing single-visit images, one with coadds, and one with difference imaging. Expected Result Actual Result Status: **Not Executed**
3	Description Verify that each of the single-visit, coadd, and difference image catalogs provide measurements in flux units. Expected Result Confirmation of measurements in catalogs encoded in flux units. Actual Result Status: **Not Executed**

5.1.3.2 LVV-T133 - Verify implementation of Provide Beam Projector Coordinate Calculation Software

Version **1**. Open *LVV-T133* test case in Jira.

Verify that the DMS provides software to calculate coordinates relating the collimated beam projector position and telescope pupil position to the illumination position on the telescope optical elements and focal plane.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description On the LSST development cluster or notebook aspect, git clone the repo containing the CBP package: `https://github.com/lsst/cbp` Expected Result Actual Result Status: **Not Executed**
2	Description Follow the steps in the package README to install the package. Expected Result Actual Result Status: **Not Executed**

3	Description Confirm that the package can be loaded in python, and that some of the tests in the 'tests/' folder will execute. Expected Result Successful execution of test scripts, which demonstrate the calculation of beam projector coordinates. Actual Result Status: **Not Executed**

5.1.3.3 LVV-T1087 - Full Table Joins Functional Test

Version **1**. Open *LVV-T1087* test case in Jira.

The objective of this test is to ensure that the full table join queries are performing as expected and establish a timing baseline benchmark for these types of queries.

Preconditions:
QSERV has been set-up following procedure at LVV-T1017.

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description

Execute query:

```
SELECT o.deepSourceId, s.objectId, s.id, o.ra, o.decl
FROM Object o, Source s WHERE o.deepSourceId=s.objectId
  AND s . flux_sinc BETWEEN 0.3 AND 0.31
```

and record execution time.

Expected Result

Query expected to run in less than 12 hours.

Actual Result

Status: **Not Executed**

2	Description
	Execute query:
	`SELECT o.deepSourceId, f.psfFlux FROM Object o, ForcedSource f`
	`WHERE o.deepSourceId=f.deepSourceId`
	`AND f . psfFlux BETWEEN 0.13 AND 0.14`
	and record execution time.
	Expected Result
	Query expected to run in less than 12 hours.
	Actual Result
	Status: **Not Executed**

5.1.3.4 LVV-T1086 - Full Table Scans Functional Test

Version **1**. Open *LVV-T1086* test case in Jira.

The objective of this test is to ensure that the full table scan queries are performing as expected and establish a timing baseline benchmark for these types of queries.

Preconditions:
QSERV has been set-up following procedure at LVV-T1017.

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Execute query: **SELECT** ra , decl , u_psfFlux , g_psfFlux , r_psfFlux **FROM** Object **WHERE** y_shapeIxx **BETWEEN** 20 **AND** 20.1 and record execution time and output size. Expected Result Query expected to run in less than 1 hour. Actual Result Status: **Not Executed**
2	Description Execute query: **SELECT** COUNT(*) **FROM** Source **WHERE** flux_sinc **BETWEEN** 1 **AND** 1.1 and record the execution time Expected Result Query expected to run in less than 12 hours.

	Actual Result
	Status: **Not Executed**
3	Description Execute query: **SELECT** COUNT(*) **FROM** ForcedSource **WHERE** psfFlux **BETWEEN** 0.1 **AND** 0.2 and record the execution time
	Expected Result Query expected to run in less than 12 hours.
	Actual Result
	Status: **Not Executed**

5.1.3.5 LVV-T1085 - Short Queries Functional Test

Version **1**. Open *LVV-T1085* test case in Jira.

The objective of this test is to ensure that the short queries are performing as expected and establish a timing baseline benchmark for these types of queries.

Preconditions:
QSERV has been set-up following procedure at LVV-T1017.

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details

1	Description Execute single object selection: **SELECT** * **FROM** Object **WHERE** deepSourceId = 9292041530376264 and record execution time. Expected Result Query runs in less than 10 seconds. Actual Result Status: **Not Executed**
2	Description Execute spatial area selection from Object: **SELECT COUNT(*) FROM** Object **WHERE** qserv_areaspec_box(316.582327, −6.839078, 316.653938, −6.781822) and record execution time. Expected Result Query runs in less than 10 seconds. Actual Result Status: **Not Executed**

5.1.3.6 LVV-T1232 - Verify Implementation of Catalog Export Formats From the Portal Aspect

Version **1**. Open *LVV-T1232* test case in Jira.

Verify that catalog data is exportable from the portal aspect in a variety of community-standard formats.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Navigate to the Portal Aspect endpoint. The stable version should be used for this test and is currently located at: https://lsst-lsp-stable.ncsa.illinois.edu/portal/app/ . Expected Result A credential-entry screen should be displayed. Actual Result Status: **Not Executed**
2	Description Enter a valid set of credentials for an LSST user with LSP access on the instance under test. Expected Result The Portal Aspect UI should be displayed following authentication. Actual Result Status: **Not Executed**
3	Description Select query type “ADQL”. Expected Result Actual Result Status: **Not Executed**
4	Description Execute the example query given in the example code below by entering the text in the ADQL Query box, then clicking “Search” at the lower left corner of the page.

Example Code

SELECT cntr, ra, decl, w1mpro_ep, w2mpro_ep, w3mpro_ep FROM wise_00.allwise_p3as_mep WHERE CONTAINS(POINT('ICRS', ra, decl), CIRCLE('ICRS', 192.85, 27.13, .2)) = 1

Expected Result

A new page will load with the search results as a table, with some plots as well.

Actual Result

Status: **Not Executed**

5	Description Click the icon that looks like a floppy disk (it says "Save the content as an IPAC, CSV, or TSV table" when you mouse over it). Expected Result Actual Result Status: **Not Executed**
6	Description • Select "CSV", then specify a destination to save the file on your local computer. • Select "VOTable", then specify a destination to save the file on your local computer. • Select "FITS", then specify a destination to save the file on your local computer. Expected Result Actual Result Status: **Not Executed**
7	Description Open each of the files (either in TOPCAT, or using Astropy io tools). Confirm that the data tables are well-formed, and that each table contains the same columns and the same number of rows. Expected Result Actual Result Status: **Not Executed**
8	Description

Currently, there is no logout mechanism on the portal.
This should be updated as the system matures.

Simply close the browser window.

Expected Result

Closed browser window. When navigating to the portal endpoint, expect to execute the steps in LVV-T849.

Actual Result

Status: **Not Executed**

5.1.3.7 LVV-T40 - Verify implementation of Generate WCS for Visit Images

Version **1**. Open *LVV-T40* test case in Jira.

Verify that Processed Visit Images produced by the AP and DRP pipelines include FITS WCS accurate to specified **astrometricAccuracy** over the bounds of the image.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify an appropriate processed dataset for this test. Expected Result A dataset with Processed Visit Images available. Actual Result

Status: **Not Executed**

2 Description

Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following:

Example Code

```
import lsst.daf.persistence as dafPersist
butler = dafPersist.Butler(inputs='DATA/path')
```

Expected Result

Butler repo available for reading.

Actual Result

Status: **Not Executed**

3 Description

Select a single visit from the dataset, and extract its WCS object and the source list.

Expected Result

A table containing detected sources, and a WCS object associated with that catalog.

Actual Result

Status: **Not Executed**

4 Description

Confirm that each CCD within the visit image contains at least **astrometricMinStandards** astrometric standards that were used in deriving the astrometric solution.

Expected Result

At least **astrometricMinStandards** from each CCD were used in determining the WCS solution.

Actual Result

Status: **Not Executed**

5 Description

Starting from the XY pixel coordinates of the sources, apply the WCS to obtain RA, Dec coordinates.

Expected Result

A list of RA, Dec coordinates for all sources in the catalog.

Actual Result

Status: **Not Executed**

6	Description We will assume that Gaia provides a source of "truth." Match the source list to Gaia DR2, and calculate the positional offset between the test data and the Gaia catalog. Expected Result A matched catalog of sources in common between the test source list and Gaia DR2. Actual Result Status: **Not Executed**
7	Description Apply appropriate cuts to extract the optimal dataset for comparison, then calculate statistics (median, 1-sigma range, etc.; also plot a histogram) of the offsets in milliarcseconds. Confirm that the offset is less than **astrometricAccuracy**. Expected Result Histogram and relevant statistics needed to confirm that the WCS transformation is accurate. Actual Result Status: **Not Executed**
8	Description Repeat Step 5, but for subregions of the image, to confirm that the accuracy criterion is met at all positions. Expected Result **astrometricAccuracy** requirement is met over the entire image. Actual Result Status: **Not Executed**

5.1.3.8 LVV-T1759 - Verify calculation of photometric outliers in gri bands

Version **1**. Open *LVV-T1759* test case in Jira.

Verify that the DM system has provided the code to calculate the photometric repeatability in the g, r, and i filters, and assess whether it meets the requirement that no more than **PF1 = 10[percent]** of the repeatability outliers exceed the outlier limit of **PA2gri = 15 millimagnitudes**.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset containing at least one field in each of the g, r, and i filters with multiple overlapping visits. Expected Result A dataset that has been ingested into a Butler repository. Actual Result Status: **Not Executed**
2	Description The 'path' that you will use depends on where you are running the science pipelines. Options: • local (newinstall.sh - based install):[path_to_installation]/loadLSST.bash • development cluster ("lsst-dev"): /software/lsstsw/stack/loadLSST.bash • LSP Notebook aspect (from a terminal): /opt/lsst/software/stack/loadLSST.bash From the command line, execute the commands below in the example code: Example Code

source ‘path‘
setup lsst_distrib

Expected Result

Science pipeline software is available for use. If additional packages are needed (for example, ‘obs’ packages such as ‘obs_subaru’), then additional ‘setup‘ commands will be necessary.

To check versions in use, type:
eups list -s

Actual Result

Status: **Not Executed**

3	Description

Execute ‘validate_drp‘ on a repository containing precursor data. Identify the path to the data, which we will call ‘DATA/path’, then execute the following (with additional flags specified as needed):

Example Code

validateDrp.py ‘DATA/path‘

Expected Result

JSON files (and associated figures) containing the Measurements and any associated “extras.”

Actual Result

Status: **Not Executed**

4	Description

Confirm that the metric PA2gri has been calculated using the threshold PF1, and that its values are reasonable.

Expected Result

A JSON file (and/or a report generated from that JSON file) demonstrating that PA2gri has been calculated (and that it used PF1).

Actual Result

Status: **Not Executed**

5.1.3.9 LVV-T1758 - Verify calculation of photometric outliers in uzy bands

Version **1**. Open *LVV-T1758* test case in Jira.

Verify that the DM system has provided the code to calculate the photometric repeatability in the u, z, and y filters, and assess whether it meets the requirement that no more than **PF1 = 10[percent]** of the repeatability outliers exceed the outlier limit of **PA2uzy = 22.5 millimagnitudes**.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset containing at least one field in each of the u, z, and y filters with multiple overlapping visits. Expected Result A dataset that has been ingested into a Butler repository. Actual Result Status: **Not Executed**
2	Description The 'path' that you will use depends on where you are running the science pipelines. Options: • local (newinstall.sh - based install):[path_to_installation]/loadLSST.bash • development cluster ("lsst-dev"): /software/lsstsw/stack/loadLSST.bash • LSP Notebook aspect (from a terminal): /opt/lsst/software/stack/loadLSST.bash From the command line, execute the commands below in the example code:

Example Code

source ‘path‘
setup lsst_distrib

Expected Result

Science pipeline software is available for use. If additional packages are needed (for example, ‘obs’ packages such as ‘obs_subaru‘), then additional ‘setup‘ commands will be necessary.

To check versions in use, type:
eups list -s

Actual Result

Status: **Not Executed**

3	Description

Execute ‘validate_drp‘ on a repository containing precursor data. Identify the path to the data, which we will call ‘DATA/path’, then execute the following (with additional flags specified as needed):

Example Code

validateDrp.py ‘DATA/path‘

Expected Result

JSON files (and associated figures) containing the Measurements and any associated “extras.”

Actual Result

Status: **Not Executed**

4	Description

Confirm that the metric PA2uzy has been calculated using the threshold PF1, and that its values are reasonable.

Expected Result

A JSON file (and/or a report generated from that JSON file) demonstrating that PA2uzy has been calculated (and that it used PF1).

Actual Result

Status: **Not Executed**

5.1.3.10 LVV-T1756 - Verify calculation of photometric repeatability in uzy filters

Version **1**. Open *LVV-T1756* test case in Jira.

Verify that the DM system has provided the code to calculate the RMS photometric repeatability of bright non-saturated unresolved point sources in the u, z, and y filters, and assess whether it meets the requirement that it shall be less than **PA1uzy = 7.5 millimagnitudes**.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset containing at least one field in each of the u, z, and y filters with multiple overlapping visits. Expected Result A dataset that has been ingested into a Butler repository. Actual Result Status: **Not Executed**
2	Description Execute 'validate_drp' on a repository containing precursor data. Identify the path to the data, which we will call 'DATA/path', then execute the following (with additional flags specified as needed): Example Code validateDrp.py 'DATA/path' Expected Result JSON files (and associated figures) containing the Measurements and any associated "extras." Actual Result Status: **Not Executed**
3	Description

	Confirm that the metric PA1uzy has been calculated, and that its values are reasonable.
	Expected Result A JSON file (and/or a report generated from that JSON file) demonstrating that PA1uzy has been calculated.
	Actual Result
	Status: **Not Executed**

5.1.3.11 LVV-T1757 - Verify calculation of photometric repeatability in gri filters

Version **1**. Open *LVV-T1757* test case in Jira.

Verify that the DM system has provided the code to calculate the RMS photometric repeatability of bright non-saturated unresolved point sources in the g, r, and i filters, and assess whether it meets the requirement that it shall be less than **PA1gri = 5.0 millimagnitudes**.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset containing at least one field in each of the g, r, and i filters with multiple overlapping visits.
	Expected Result A dataset that has been ingested into a Butler repository.
	Actual Result

	Status: **Not Executed**
2	Description Execute 'validate_drp' on a repository containing precursor data. Identify the path to the data, which we will call 'DATA/path', then execute the following (with additional flags specified as needed): Example Code validateDrp.py 'DATA/path' Expected Result JSON files (and associated figures) containing the Measurements and any associated "extras." Actual Result Status: **Not Executed**
3	Description Confirm that the metric PA1gri has been calculated, and that its values are reasonable. Expected Result A JSON file (and/or a report generated from that JSON file) demonstrating that PA1gri has been calculated. Actual Result Status: **Not Executed**

5.1.3.12 LVV-T125 - Verify implementation of Simulated Data

Version **1**. Open *LVV-T125* test case in Jira.

Verify that the DMS can inject simulated data into data products for testing.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset that has been (or can be readily) processed through single-frame processing and coaddition. Expected Result The ‘calexp‘ and ‘deepCoadd_calexp‘ images and their associated source catalogs are created. Actual Result Status: **Not Executed**
2	Description Roughly determine the coordinates of a bounding box that is contained within the images that were processed. Expected Result RA, Dec boundaries of a region in which to generate fake sources. Actual Result Status: **Not Executed**
3	Description Generate a catalog in the correct format for ‘insertFakes‘ to accept. The catalog should specify positions and magnitudes of stars (and optionally, parameters specifying galaxy shape, if galaxies are also being inserted). Expected Result An input catalog of fake source positions and magnitudes to be inserted into the images. Actual Result Status: **Not Executed**
4	Description Execute ‘insertFakes.py‘ on the repository, specifying the input catalog from the previous step. Expected Result A repository with images that have fake sources inserted. Actual Result Status: **Not Executed**
5	Description

	Run 'multiBandDriver.py' on the repository, specifying the fake-source repository as the input. Expected Result 'calexp' and coadd images containing the artificial sources and sources catalogs that contain their measurements along with the sources detected in the original run. Actual Result Status: **Not Executed**
6	Description Confirm that the injected sources appear in the images and the catalogs. Expected Result Fake sources and their measured properties are recoverable. Actual Result Status: **Not Executed**

5.1.3.13 LVV-T36 - Verify implementation of Difference Exposures

Version **1**. Open *LVV-T36* test case in Jira.

Verify successful creation of a
1. PSF-matched template image for a given Processed Visit Image
2. Difference Exposure from each Processed Visit Image

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description The 'path' that you will use depends on where you are running the science pipelines. Options: • local (newinstall.sh - based install):[path_to_installation]/loadLSST.bash • development cluster ("lsst-dev"): /software/lsstsw/stack/loadLSST.bash • LSP Notebook aspect (from a terminal): /opt/lsst/software/stack/loadLSST.bash From the command line, execute the commands below in the example code: Example Code source 'path' setup lsst_distrib Expected Result Science pipeline software is available for use. If additional packages are needed (for example, 'obs' packages such as 'obs_subaru'), then additional 'setup' commands will be necessary. To check versions in use, type: eups list -s Actual Result Status: **Not Executed**
2	Description Perform the steps of Alert Production (including, but not necessarily limited to, single frame processing, ISR, source detection/measurement, PSF estimation, photometric and astrometric calibration, difference imaging, DIASource detection/measurement, source association). During Operations, it is presumed that these are automated for a given dataset. Expected Result An output dataset including difference images and DIASource and DIAObject measurements. Actual Result Status: **Not Executed**
3	Description

Verify that the expected data products have been produced, and that catalogs contain reasonable values for measured quantities of interest.

Expected Result

Actual Result

Status: **Not Executed**

4	Description 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. Expected Result Actual Result Status: **Not Executed**

5.1.3.14 LVV-T126 - Verify implementation of Image Differencing

Version **1**. Open *LVV-T126* test case in Jira.

Verify that the DMS can performance image differencing from single exposures and coadds.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a repository containing data that have been processed through the difference imaging pipeline. (e.g., the HiTS 2015 data that are processed monthly for testing) Expected Result A dataset containing calexps, difference images, and source catalogs (of diaSrcs). Actual Result Status: **Not Executed**

2	Description
	Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following:
	Example Code
	`import lsst.daf.persistence as dafPersist`
	`butler = dafPersist.Butler(inputs='DATA/path')`
	Expected Result
	Butler repo available for reading.
	Actual Result
	
	Status: **Not Executed**

3	Description Extract a 'calexp', a 'deepDiff_differenceExp', and the 'deepDiff_diaSrc' catalog of measurements. Expected Result Well-formed images and catalogs containing the calexp from the visit image and the difference image, and measurements of sources from the difference image. Actual Result Status: **Not Executed**
4	Description Confirm (by visual inspection) that the difference image is mostly blank sky (i.e., has had a template of the same field subtracted), and that the source catalog contains sources with photometric and astrometric measurements. Expected Result A mostly blank image (with perhaps some artifacts due to imperfect subtraction) and a catalog of sources detected/measured from that image.

Actual Result

Status: **Not Executed**

5.1.3.15 LVV-T39 - Verify implementation of Generate Photometric Zeropoint for Visit Image

Version **1**. Open *LVV-T39* test case in Jira.

Verify that Processed Visit Image data products produced by the DRP and AP pipelines include the parameters of a model that relates the observed flux on the image to physical flux units.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify a dataset with processed visit images in multiple filters. Expected Result Actual Result Status: **Not Executed**
2	Description Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following: Example Code

```
import lsst.daf.persistence as dafPersist
butler = dafPersist.Butler(inputs='DATA/path')
```

Expected Result

Butler repo available for reading.

Actual Result

Status: **Not Executed**

3	Description Extract the photometric zeropoint from the source catalog associated with a visit image. Repeat this for all available filters, and confirm that the zeropoint has been set, and has a reasonable value. Expected Result A zeropoint that enables one to convert the measured fluxes to magnitudes. Actual Result Status: **Not Executed**
4	Description Extract fluxes for some sources, and convert them to magnitudes. Confirm that the distribution spans a reasonable range. Expected Result In most cases, well-measured magnitudes (i.e., for high S/N measurements) should be between 12 to 28 for all bands. Actual Result Status: **Not Executed**

5.1.3.16 LVV-T46 - Verify implementation of Prompt Processing Performance Report Definition

Version **1**. Open *LVV-T46* test case in Jira.

Verify that the DMS produces a Prompt Processing Performance Report. Specifically check that the number of observations that describe each of the following:
1. Successfully processed, recoverable failures, unrecoverable failures.

2. Archived
3. Result in science.

This is testing more the processing rather than the observatory system.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Execute single-day operations rehearsal, observe report
	Expected Result
	Actual Result
	Status: **Not Executed**

5.1.3.17 LVV-T38 - Verify implementation of Processed Visit Images

Version **1**. Open *LVV-T38* test case in Jira.

Verify that the DMS
1. Successfully produces Processed Visit Images, where the instrument signature has been removed.
2. Successfully combines images obtained during a standard visit.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Identify suitable precursor datasets containing unprocessed raw images. Expected Result Actual Result Status: **Not Executed**
2	Description Run the Prompt Processing payload on these data. Verify that Processed Visit Images are generated at correct size and with significant instrumental artifacts removed. Expected Result Raw precursor dataset images have been processed into Processed Visit Images, with instrumental artifacts corrected. Actual Result Status: **Not Executed**
3	Description Run camera test stand data through full acquisition+backbone+ISR. Expected Result Actual Result Status: **Not Executed**
4	Description

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.

Expected Result

Raw images have been processed into Processed Visit Images, with instrumental artifacts corrected.

Actual Result

Status: **Not Executed**

5.1.3.18 LVV-T42 - Verify implementation of Processed Visit Image Content

Version **1**. Open *LVV-T42* test case in Jira.

Verify that Processed Visit Images produced by the DRP and AP pipelines include the observed data, a mask array, a variance array, a PSF model, and a WCS model.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details

1	Description
	Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following:
	Example Code
	`import lsst.daf.persistence as dafPersist`
	`butler = dafPersist.Butler(inputs='DATA/path')`
	Expected Result

Butler repo available for reading.

Actual Result

Status: **Not Executed**

2 Description

Ingest the data from an appropriate processed dataset.

Expected Result

Actual Result

Status: **Not Executed**

3 Description

Select a single visit from the dataset, and extract its WCS object, calexp image, psf model, and source list.

Expected Result

Actual Result

Status: **Not Executed**

4 Description

Inspect the calexp image to ensure that

1. A well-formed image is present,
2. The variance plane is present and well-behaved,
3. Mask planes are present and contain information about defects.

Expected Result

An astronomical image with mask and variance planes. This can be readily visualized using Firefly, which displays mask planes by default.

Actual Result

Status: **Not Executed**

5 Description

Plot images of the PSF model at various points, and verify that the PSF differs with position.

Expected Result

A “star-like” image of the PSF evaluated at various positions. The PSF should vary slightly with position (this could be readily visualized by taking a difference of PSFs at two positions).

	Actual Result
	Status: **Not Executed**
6	Description Starting from the XY pixel coordinates of the sources, apply the WCS to obtain RA, Dec coordinates. Plot these positions and confirm that they match the expected values from the WCS object.
	Expected Result RA, Dec coordinates that are returned should be near the central position of the visit coordinate as given in either the calexp metadata or the WCS.
	Actual Result
	Status: **Not Executed**
7	Description Repeat steps 2-6, but now with difference images created by the Alert Production pipeline (for example, in the 'ap_verify' test data processing).
	Expected Result
	Actual Result
	Status: **Not Executed**

5.1.3.19 LVV-T149 - Verify implementation of Catalog Queries

Version **1**. Open *LVV-T149* test case in Jira.

Verify that SQL, or a similar structured language, can be used to query catalogs.

Preconditions:
An operational QSERV database that has been verified via LVV-T1085 and LVV-T1086 and LVV-T1087.

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Execute a simple query (for example, the one below) and confirm that it returns the expected result. Example Code SELECT * FROM Object WHERE qserv_areaspec_box(316.582327, −6.839078, 316.653938, −6.781822) Expected Result A catalog of objects satisfying the specified constraints. Actual Result Status: **Not Executed**
2	Description Repeat the query from all available access routes (e.g., an external VO client, internal DM tools on the development cluster, the Science Platform query tool, and from within the Notebook Aspect), confirming in each case that the results are as expected. Expected Result Actual Result Status: **Not Executed**

5.1.3.20 LVV-T151 - Verify Implementation of Catalog Export Formats From the Notebook Aspect

Version **1**. Open *LVV-T151* test case in Jira.

Verify that catalog data is exportable from the notebook aspect in a variety of community-standard formats.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Authenticate to the notebook aspect of the LSST Science Platform (NB-LSP). This is currently at https://lsst-lsp-stable.ncsa.illinois.edu/nb. Expected Result Redirection to the spawner page of the NB-LSP allowing selection of the containerized stack version and machine flavor. Actual Result Status: **Not Executed**
2	Description Spawn a container by: 1) choosing an appropriate stack version: e.g. the latest weekly. 2) choosing an appropriate machine flavor: e.g. medium 3) click “Spawn” Expected Result Redirection to the JupyterLab environment served from the chosen container containing the correct stack version. Actual Result Status: **Not Executed**
3	Description Open a new launcher by navigating in the top menu bar “File” -> “New Launcher” Expected Result A launcher window with several sections, potentially with several kernel versions for each. Actual Result

Status: **Not Executed**

4 Description

Select the option under “Notebook” labeled “LSST” by clicking on the icon.

Expected Result

An empty notebook with a single empty cell. The kernel show up as “LSST” in the top right of the notebook.

Actual Result

Status: **Not Executed**

5 Description

Execute a query in a notebook to select a small number of stars. In the example code below, we query the WISE catalog, then extract the results to an Astropy table.

Example Code

```
import pandas
import pyvo
service = pyvo.dal.TAPService('http://lsst-lsp-stable.ncsa.illinois.edu/api/tap')

results = service.search(“SELECT ra, decl, w1mpro_ep, w2mpro_ep, w3mpro_ep FROM wise_00.allwise_p3as_mep WHERE CONTAINS(POINT(‘ICRS’, ra, decl), CIRCLE(‘ICRS’, 192.85, 27.13, .2)) = 1”)
tab = results.to_table()
```

Expected Result

Actual Result

Status: **Not Executed**

6 Description

Using the example code below, save the files to your storage space on the LSP Notebook Aspect.

Confirm that non-empty output files appear on disk.

Example Code

```
tab.write(‘test.csv’, format=‘ascii.csv’)
tab.write(‘test.vot’, format=‘votable’)
tab.write(‘test.fits’, format=‘fits’)
```

Expected Result

For the example given here, there should be the following files with the file size as listed:

- test.csv 5.7M
- test.vot 16M
- test.fits 4.5M

Actual Result

Status: **Not Executed**

7 Description

Check that these files contain the same number of rows:

Example Code

```
from astropy.table import Table
dat_csv = Table.read('test.csv', format='ascii.csv')
dat_vot = Table.read('test.vot', format='votable')
dat_fits = Table.read('test.fits', format='fits')


import numpy as np
print(np.size(dat_csv), np.size(dat_vot), np.size(dat_fits))
```

Expected Result

Print statement produces output "97058 97058 97058".

Actual Result

Status: **Not Executed**

8 Description

Under the 'File' menu at the top of your Jupyter notebook session, select one of the following:

- Save All, Exit, and Log Out
- Exit and Log Out Without Saving

Expected Result

You will be returned to the LSP landing page: `https://lsst-lsp-stable.ncsa.illinois.edu/` It is now safe to close the browser window.

Actual Result

Status: **Not Executed**

5.1.3.21 LVV-T45 - Verify implementation of Prompt Processing Data Quality Report Definition

Version **1**. Open *LVV-T45* test case in Jira.

Verify that the DMS produces a Prompt Processing Data Quality Report. Specifically check absolute value and temporal variation of
1. Photometric zeropoint
2. Sky brightness
3. Seeing
4. PSF
5. Detection efficiency

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Ingest raw data from L1 Test Stand DAQ. Expected Result Actual Result Status: **Not Executed**
2	Description Perform the steps of Alert Production (including, but not necessarily limited to, single frame processing, ISR, source detection/measurement, PSF estimation, photometric and astrometric calibration, difference imaging, DIASource detection/measurement, source association). During Operations, it is presumed that these are automated for a given dataset.

Expected Result

An output dataset including difference images and DIASource and DIAObject measurements.

Actual Result

Status: **Not Executed**

3	Description Verify that the expected data products have been produced, and that catalogs contain reasonable values for measured quantities of interest. Expected Result Actual Result Status: **Not Executed**
4	Description Load the Prompt Processing QC reports, and observe that a dynamically updated Data Quality Report has become available at the relevant UI. Expected Result A Prompt Processing QC report is available via a UI, and contains information about the photometric zeropoint, sky brightness, seeing, PSF, and detection efficiency, and possibly other relevant quantities. Actual Result Status: **Not Executed**
5	Description Check that a static report is created and archived in a readily-accessible location. Expected Result Persistence of a static QC report in an accessible location, containing the same information as in the report from Step 3. Actual Result Status: **Not Executed**

5.1.3.22 LVV-T146 - Verify implementation of DMS Initialization Component

Version **1**. Open *LVV-T146* test case in Jira.

Demonstrate that the DMS can be initialized in a safe state that will not allow data corruption/loss.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Power-cycle all of the DM systems at each Facility. Expected Result Restart of all DM systems. Actual Result Status: **Not Executed**
2	Description Observe each system and ensure that it has recovered in a properly initialized state. Expected Result Systems are all active and initialized for their designated purpose. Actual Result Status: **Not Executed**

5.1.3.23 LVV-T144 - Verify implementation of Task Specification

Version **1**. Open *LVV-T144* test case in Jira.

Verify that the DMS provides the ability to define a new or modified pipeline task without recompilation.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Inspect software architecture. Verify that there exist Tasks that can be run and configured without re-compilation. Expected Result Confirmation that the software architecture has allowed for reconfiguring and running Tasks without recompilation. Actual Result Status: **Not Executed**
2	Description 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. Expected Result Successful Task execution with different configurations, including confirmation that the outputs are different from tasks with altered configurations. Actual Result Status: **Not Executed**

5.1.3.24 LVV-T145 - Verify implementation of Task Configuration

Version **1**. Open *LVV-T145* test case in Jira.

Verify that the DMS software provides configuration control to define, override, and verify the configuration for a DMS Task.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Inspect software design to verify that one can define the configuration for a Task. Expected Result Actual Result Status: **Not Executed**
2	Description Run a Task with a known invalid configuration. Verify that the error is caught before the science algorithm executes. Expected Result Actual Result Status: **Not Executed**
3	Description

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.

Expected Result

Actual Result

Status: **Not Executed**

5.1.3.25 LVV-T1264 - Verify implementation of archiving camera test data

Version **1**. Open *LVV-T1264* test case in Jira.

Verify that a subset of camera test data has been ingested into Butler repos and is available through standard data access tools.

Preconditions:

Execution status: **Not Executed**

Final comment:

Detailed steps results:

Step	Step Details
1	Description Obtain some data on a camera test stand. Expected Result Actual Result Status: **Not Executed**

2	Description Wait a sufficient amount of time, then confirm that automatic transfer/ingest of the data has occurred, and a repo is available at NCSA. Expected Result The data is present at NCSA in non-empty repos. Actual Result Status: **Not Executed**
3	Description Identify the relevant Butler repo of ingested camera test stand data. Expected Result Actual Result Status: **Not Executed**

4	Description
	Identify the path to the data repository, which we will refer to as 'DATA/path', then execute the following:
	Example Code
	`import lsst.daf.persistence as dafPersist`
	`butler = dafPersist.Butler(inputs='DATA/path')`
	Expected Result
	Butler repo available for reading.
	Actual Result
	Status: **Not Executed**

5	Description Read various repo data products with the Butler, and confirm that they contain the expected data. Expected Result Camera test stand data that is well-formed. Actual Result Status: **Not Executed**

A Acronyms used in this document

Acronym	Description
ADC	Analogue-to-Digital Converter
ADQL	Astronomical Data Query Language
AP	Alert Production
CBP	Collimated Beam Projector
CCD	Charge-Coupled Device
CSV	Comma Separated Values
DAQ	Data Acquisition System
DM	Data Management
DMS	Data Management Subsystem
DMS-REQ	Data Management System Requirements prefix
DMSR	DM System Requirements; LSE-61
DMTR	DM Test Report
DRP	Data Release Production
FITS	Flexible Image Transport System
HSC	Hyper Suprime-Cam
IPAC	No longer an acronym; science and data center at Caltech
ISR	Instrument Signal Removal
JSON	JavaScript Object Notation
L1	Lens 1
LDF	LSST Data Facility
LDM	LSST Data Management (Document Handle)
LSE	LSST Systems Engineering (Document Handle)
LSP	LSST Science Platform (now Rubin Science Platform)
LSST	Legacy Survey of Space and Time (formerly Large Synoptic Survey Telescope)
NCSA	National Center for Supercomputing Applications
PDF	Portable Document Format
PMCS	Project Management Controls System
PSF	Point Spread Function
QC	Quality Control
RA	Right Ascension
RMS	Root-Mean-Square

SQL	Structured Query Language
TOPCAT	Tool for OPerations on Catalogues And Tables
UI	User Interface
VO	Virtual Observatory
WCS	World Coordinate System
WISE	Wide-field Survey Explorer

B Traceability

Test Case	VE Key	VE Summary
LVV-T28	LVV-178	DMS-REQ-0347-V-01: Measurements in catalogs
LVV-T36	LVV-7	DMS-REQ-0010-V-01: Difference Exposures
LVV-T38	LVV-29	DMS-REQ-0069-V-01: Processed Visit Images
LVV-T39	LVV-12	DMS-REQ-0029-V-01: Generate Photometric Zeropoint for Visit Image
LVV-T40	LVV-13	DMS-REQ-0030-V-01: Absolute accuracy of WCS
LVV-T42	LVV-31	DMS-REQ-0072-V-01: Processed Visit Image Content
LVV-T45	LVV-39	DMS-REQ-0097-V-01: Level 1 Data Quality Report Definition
LVV-T46	LVV-41	DMS-REQ-0099-V-01: Level 1 Performance Report Definition
LVV-T125	LVV-6	DMS-REQ-0009-V-01: Simulated Data
LVV-T126	LVV-14	DMS-REQ-0032-V-01: Image Differencing
LVV-T133	LVV-182	DMS-REQ-0351-V-01: Provide Beam Projector Coordinate Calculation Software
LVV-T144	LVV-136	DMS-REQ-0305-V-01: Task Specification
LVV-T145	LVV-137	DMS-REQ-0306-V-01: Task Configuration
LVV-T146	LVV-128	DMS-REQ-0297-V-01: DMS Initialization Component
LVV-T149	LVV-33	DMS-REQ-0075-V-01: Catalog Queries
LVV-T151	LVV-35	DMS-REQ-0078-V-01: Catalog Export Formats
LVV-T1085	LVV-33	DMS-REQ-0075-V-01: Catalog Queries
	LVV-9787	DMS-REQ-0356-V-04: Max time to retrieve low-volume query results
LVV-T1086	LVV-33	DMS-REQ-0075-V-01: Catalog Queries
	LVV-188	DMS-REQ-0357-V-01: Result latency for high-volume full-sky queries on the Object table
	LVV-185	DMS-REQ-0354-V-01: Result latency for high-volume complex queries
LVV-T1087	LVV-33	DMS-REQ-0075-V-01: Catalog Queries
	LVV-185	DMS-REQ-0354-V-01: Result latency for high-volume complex queries

LVV-T1232	LVV-35	DMS-REQ-0078-V-01: Catalog Export Formats
LVV-T1264	LVV-9637	DMS-REQ-0372-V-01: Archiving Camera Test Data
LVV-T1756	LVV-3401	DMS-REQ-0359-V-01: RMS photometric repeatability in uzy
LVV-T1757	LVV-9759	DMS-REQ-0359-V-10: RMS photometric repeatability in gri
LVV-T1758	LVV-9758	DMS-REQ-0359-V-09: Repeatability outlier limit in uzy
	LVV-9752	DMS-REQ-0359-V-03: Max fraction of outliers among non-saturated sources
LVV-T1759	LVV-9752	DMS-REQ-0359-V-03: Max fraction of outliers among non-saturated sources
	LVV-9754	DMS-REQ-0359-V-05: Repeatability outlier limit in gri