

DRAFT Last revised 12 Feb 2009

DC3 Verification and Validation Plan

Introduction

Caveat

Data Challenge 3 is introducing the formal specification of Baseline Project Plans. Not every requirement stated in LSST DM Software Development Process has been implemented. Those procedures not possible or required for DC3 will be noted in the following text.

As required by the Lsst SDP Verification and Validation Guidelines, this DC3 Verification and Validation Plan checks products against their specifications by ^[1]

- checking that each software item meets specified requirements;
- checking each software item before it is used as an input to another activity;
- ensuring that the amount of verification and validation effort is adequate to show each software item is suitable for operational use.

This document is a member of the DC3 Project Plans defining

- Software Release Project Management;
- Configuration Management;
- Verification and Validation (this document); and
- Software Quality Assurance.

Purpose of Document

This document defines

- the processes which validate that the DC3 code implements the DC3 design; and
- the processes which verify that the above mentioned validation processes have been properly followed.

Baseline Identifier

The baseline identifier for Data Challenge 3 is **DC3**.

Definitions, acronyms and abbreviations

- **DC3**: abbreviation for LSST DM Data Challenge 3.
- **SDP**: abbreviation for the Software Development Plan.

References

- [1] "ESA Guide to software verification and validation", PSS-05-10, ESA 1995, [?ftp://ftp.estec.esa.nl/pub/wm/wme/bssc/PSS0510.pdf](ftp://ftp.estec.esa.nl/pub/wm/wme/bssc/PSS0510.pdf), [?Local Copy](#)

Reviews

Inspection

An automatic coding standards checker should be available prior to the completion of DC3. However, in advance of that tool's acquisition and its retrofitting to include the rules from the DM Coding Language Standards, the implementation code is not being consistently monitored for compliance to the DM Coding Language Standards.

Technical Review Process

DC3 is a mid-point in a iterative development cycle of 5 Data Challenges. The challenges were defined at the start of the project to incrementally prototype the data management software such that the software framework and algorithms iteratively extended capabilities whilst benchmarking performance. As such, the completed LSST Project-wide reviews for the Science Requirements, the DM Functional Requirements, and the Conceptual Design Review reflect the successful technical review of DC3 for those phases.

A formal DC3 Preliminary Design Review for DC3 did not occur. The Preliminary Design developed for the full LSST DM model was used as a base and refined at focussed meetings of DC3 Applications and Middleware working groups.

A formal DC3 Detail Design Review did not occur. Many of the detailed designs for the DC3 Applications objects were developed during an intensive DC3 design workshop lead by the ICONIX developer and co-chaired by the initial DM Architecture team.

The DC3 Final Report will be considered the DC3 Release Review.

Tracing

The direct link between User Requirements to validated code is traced using the SparxSystem Enterprise Architect Tool:

System Engineering Level (governed by LSST System Engineering Management Plan) trace from

- SRD Requirements in the SysML model which have derived
- LSST System Requirements in the SysML model which have derived
- DM Subsystem Requirements in the SysML model which are satisfied by
- DM Subsystem Blocks (WBS elements) in the SysML model which are realized by the
- Software Engineering Level (governed by Software Development Plan).

DM Tracing continues unbroken from the Software Engineering Level (governed by Software Development Plan) to

- DM Use Cases in the UML model which are realized by
- Robustness (control, boundary, and entity/domain) objects in the UML model which are implemented by
- Classes in the UML model which are implemented as
- Code modules in the code repository and in the UML model.

Validation Testing of the DC3b Data Products

User Level Acceptance Validation

User Level Acceptance Validation is performed by Science Collaborators using DC3b data results in pursuit of their scientific interests. Within 2 months of acquiring the data, they should provide a report to the DM Scientist discussing such topics as:

- how did you use the data to do your science?
- what problems did you encounter?
- what auxiliary data products were missing?
- what data quality issues did you find?
- what should we change before LSST goes live?

It would be especially useful if the Scientist would also provide the tools developed to analyze the DC3b data for possible inclusion in DM's SQA validation toolkit.

In addition to the summary report, the Science Collaborator may want to provide immediate feedback to the DM Scientist on issues relating to format, use, quality, accuracy, reliability, etc.

Science Data Processing Validation

Science Data Processing Validation is performed by Data Management scientists, including support from the IPAC LSST scientists, and is coordinated by the DM Scientist. Each major data deliverable will be validated, independently from an algorithm's developer, for quality metrics appropriate for those data.

The Science Data Quality Assessment System (SDQA) is an automated software system which collects, performs statistical analyses and presents information related to the quality of the science data. It is designed to be able to analyze the quality of the data collected and processed in operations and to stay on top of the data rate and survey progress.

The SDQA system will help assess the quality of processing during DC3b. The automated tests will commence once both the DC3b SDQA software and the DC3b software which generates the quality metrics are available.

Data Management scientists can utilize this system as SDQA analysts to identify issues and provide feedback to the DM Scientist, if and when issues are exposed. The DM Scientist is responsible for either resolving the issues prior to DC3b completion or ensuring the unresolved issues are retained until fixed in a later release. Status reports on the SDQA progress on each deliverable should be provided in the weekly Applications Working Group meeting.

Tests and workflow processes developed whilst validating a data deliverable should be documented. Analysis tools (e.g. python scripts, matlab routines, database queries) created to assist the analysis should be archived within the LSST source archive. A status report on the results of the analysis should be provided to the DM Scientist.

The major DC3b data deliverables are enumerated in Tim Axelrod's DC3b Data Quality Requirements document:

- [?http://dev.lsstcorp.org/trac/attachments/wiki/DC3bManagement/DC3b_Data_Quality_Requirements.pdf](http://dev.lsstcorp.org/trac/attachments/wiki/DC3bManagement/DC3b_Data_Quality_Requirements.pdf)

A secondary version of that file including additional comments by DM developers is available at:

- [?http://dev.lsstcorp.org/trac/wiki/DC3bManagement](http://dev.lsstcorp.org/trac/wiki/DC3bManagement)

Unit Testing

Unit Testing is performed by each developer on each module which s/he has either implemented or modified. Units tests should validate that a module performs according to implementation specifications and is reliable, consistent and accurate. The unit tests should be archived within the LSST source archive.

Unit tests should successfully complete without error prior to modifications being merged from the development area (generally a Ticket branch) into the LSST archive 'trunk' branch.

Unit tests are automatically run daily and failures are immediately emailed to the responsible developer.

Test Plan Overviews

- Acceptance Test Plan Overview - not applicable to DC3.
- System Test Plan Overview - not applicable to DC3.
- [DC3a Integration Test Plan Overview](#)
- DC3b Integration Test Plan Overview? to be created later.....
- **TBD** Unit Test Plan Overview.