SCALE Procedure for Feature Changes

Prepared by
B. T. Rearden and J. P. Lefebvre
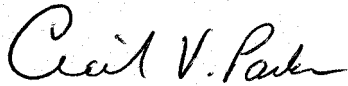
Reactor and Nuclear Systems Division (RNSD)
Oak Ridge National Laboratory

Date Revised: May 22, 2013

Approvals:

_____   _____
SCALE Project Leader   Date

_____   _____
Director, Reactor and Nuclear Systems Division   Date

_____   _____
Quality Manager, Performance Analysis and Quality   Date

SCALE Procedure for Feature Changes

1.0   PURPOSE

To describe the methods for updating, testing and documenting changes to SCALE code system.   The procedure outlined in this document complies with the Configuration Management Plan (CMP) for the SCALE code system, SCALE-CMP-001, which is controlled by the Reactor and Nuclear Systems Division (RNSD).

2.0   SCOPE

This procedure will be applied any time corrections or enhancements are to be made to any of the software or data features in the SCALE code system.

3.0   PROCEDURE

3.1   A SCALE Quality Assurance Feature Case in the electronic tracking system must be completed any time a change is made to SCALE, whether it is a correction or an enhancement. The Kanban process is used to track the progress of each Feature through design, implementation, documentation, testing, and deployment. The Kanban steps are: Proposed, Approved, In Progress, In Testing, Ready to Ship, and Deployed. If there is a code discrepancy, a SCALE Discrepancy Report (SDR) is also required. Refer to SCALE-CMP-001 for the definition of a discrepancy and the resolution and reporting procedure.

3.2   **Proposed** - The Project Leader, developers, or users may request changes to the SCALE software configuration by initializing a SCALE Quality Assurance Feature Case in the electronic tracking system. Proposed features are submitted by the requester and assigned to the Project Leader in the electronic tracking system. The Project Leader places the Feature to the Proposed state in Kanban, designates a Technical Reviewer for the Case, and assigns the Case to a Developer. The Developer reviews the high-level feature request, coordinates with other members of the development team to form the initial design and testing strategy for the Feature. The Developer records the following information in the Case and assigns it to the Project Leader:

   3.2.1   Functional Specification - Description of user-level functionality of this Feature (e.g. new capabilities or improved performance) and an overview of user input for accessing the new Feature (e.g. new keywords or input blocks)

   3.2.2   Functional Tests - Description of tests to verify user-level operability and accuracy of this Feature (e.g. test cases)

   3.2.3   Implementation Details - Description of the actual work performed. May include algorithms, flow diagrams, data structures, etc.

   3.2.4   Implementation Testing - Description of code and data level testing that may not be available at the user level (e.g. unit tests)

   3.2.5   Documentation - Description of user and developer documentation of the

Feature. May include updates to the software manual, in-line comments, Doxygen markup or other forms of documentation

3.2.6   Charge Number – ORNL billing number to be used for work related to this Feature

3.2.7   Sponsor – Name of sponsor for this work

3.2.8   Milestone – SCALE milestone where this Feature is expected to be deployed

3.2.9   Estimate – Amount of labor effort in units of time that is expected to design, implement, test, document, and deploy this Feature

Using a graded approach, the Project Leader and Technical Reviewer review the planned implementation and iterate with the development team as needed to form a consensus set of requirements.

3.3   **Approved** - The Project Leader signifies approval of the design requirements by changing the Kanban status of the Feature from Proposed to Approved. The feature request is assigned a SCALE Change Log (SCL) number and docketed by the SQA Coordinator, and the Feature Case assigned to the Developer. The docket number begins with SCL and is assigned a sequence number in the form *SCL-YYYY-NNN*. *YYYY* equals the current fiscal year and *NNN* equals a number beginning with *001* and increasing by one each time an SCL identifier is assigned. The complete sequence of numbering restarts at the beginning of each fiscal year.

3.4   **In Progress** - The Developer notifies the Project Leader when work begins on the Feature implementation, and the Project Leader changes the Kanban status to In Progress. The development team implements the Feature, testing and documentation according to the design requirements. The Developer may revise the requirements during the development process, with the approval of the Project Leader and Technical Reviewer. As work progresses, the development team pushes feature updates and tests to the SCALE Development repository. Any changes to the repository are cross-referenced in the Feature Case, and user documentation of the feature is attached to the Case. Interim reviews by a Technical Reviewer or Code Reviewer may be requested by the development team or Project Leader throughout the In Progress state of the Feature.

The development team ensures that the implementation of the Feature does not compromise the functionality of existing features without prior authorization from the Project Leader. The performance of existing features is quantified with existing regression tests, unit tests, and sample problems, collectively called the Test Suite. As the Feature is implemented, the Test Suite is amended to capture the performance of the new Feature, and the new tests are attached to the Feature Case. The Test Suite is reviewed by the development team to ensure that all new and existing features perform as expected on all designated platforms.

When the Feature, tests, and documentation are completed and recorded in the Case, the Developer notifies the Project Leader that the Feature is available for

testing.

3.5   **In Testing** - When the Developer notifies the Project Leader that a Feature is complete, and the Project Leader updates the Kanban status to In Testing. To begin the final testing of the Feature, the entire Test Suite must perform as expected on all designated platforms in the Development repository. The SQA Coordinator pushes all updates to the SCALE Staging repository, builds SCALE, and ensures that the Test Suite performs as expected on all designated platforms. When the Feature is available in the Staging area, the SQA Coordinator assigns the Feature Case to the Technical Reviewer.

The Technical Reviewer tests the Feature to ensure its functionality and reviews the documentation for completeness and clarity, using a graded approach. The Technical Reviewer should consider the following items in the review:

3.5.1   The technical validity and completeness per the code design requirements
3.5.2   The adequacy of the verification and/or validation
3.5.3   The adequacy of the documentation and its consistency with the existing system documentation
3.5.4   The adequacy and the completeness of the sample problems for inclusion in the publicly released software version
3.5.5   The adequacy of regression and unit tests to quantify the performance of the feature relative to other SCALE development activities.

The Technical Reviewer may create additional tests that were not contributed by the Developer. The Technical Reviewer or Project Leader may request additional inspection by one or more Code Reviewers and/or additional Technical Reviewers. Any issues identified by the reviewers are presented to the Developer for resolution. If a consensus resolution cannot be achieved, the Project Leader is notified and will determine the course of action. Any modification identified during the In Testing phase is implemented in the Development repository and the SQA Coordinator repeats the previous action of pushing, building, and testing the modified Feature in the Staging repository. When the Reviewers are confident that the feature is fully functional and well documented, the Technical Reviewer assigns the Feature Case to the Project Leader. The Project Leader performs an assessment of the Feature, testing, documentation, and review. The Project Leader enters QA comments into the Case to succinctly describe the new Feature.

3.6   **Ready to Ship** - The Feature is designated as Ready to Ship when the Project Leader completes his final review and assigns the Case to the SQA Coordinator. When all Features currently under review in the Staging repository are approved by the Project Leader, the SQA Coordinator pushes the Staging repository to the Production repository, builds SCALE, and ensures that the Test Suite performs as expected on all designated platforms. Only features that are Ready to Ship will be available in the Production repository.

3.7 **Deployed** - When the Project Leader determines that sufficient features are available for a beta release or a general release, the SQA Coordinator creates a Tag of the Production repository, builds SCALE, and ensures that the Test Suite performs as expected on all designated platforms. An installer is created and SCALE is deployed. At this point, all Features included in the release are designated as Deployed.