

**Lawrence Livermore National Laboratory**

**NARAC Software Quality Assurance  
SQA Program Effectiveness Review**



**Hoyt Walker**

***February 22, 2010***

LLNL-PRES-425372

# Outline

- Risk Level Grading
- SQA Activities
- SQA Guidance



# Risk Level Grading

- LLNL Institutional Software Quality Assurance Program (ISQAP) Risk-Based, Graded Approach
- Risk Consequence Assessment – Tier 3
  - Tier 3 for Performance, Political/Public Perception, Security
  - Tier 4 for ES&H
  - Determination based on NARAC Management review
- Likelihood of Failure Assessment – 3.9
- Risk Level 4



# SQA Work Activities

1. Software quality program management & planning
2. Software risk management
3. Software configuration management
4. Procurement and supplier management
5. Software requirements identification & management
6. Software design and implementation
7. Software safety
8. Verification & validation
9. Problem reporting and corrective action
10. Training



# 1. Software Quality Program Management & Planning

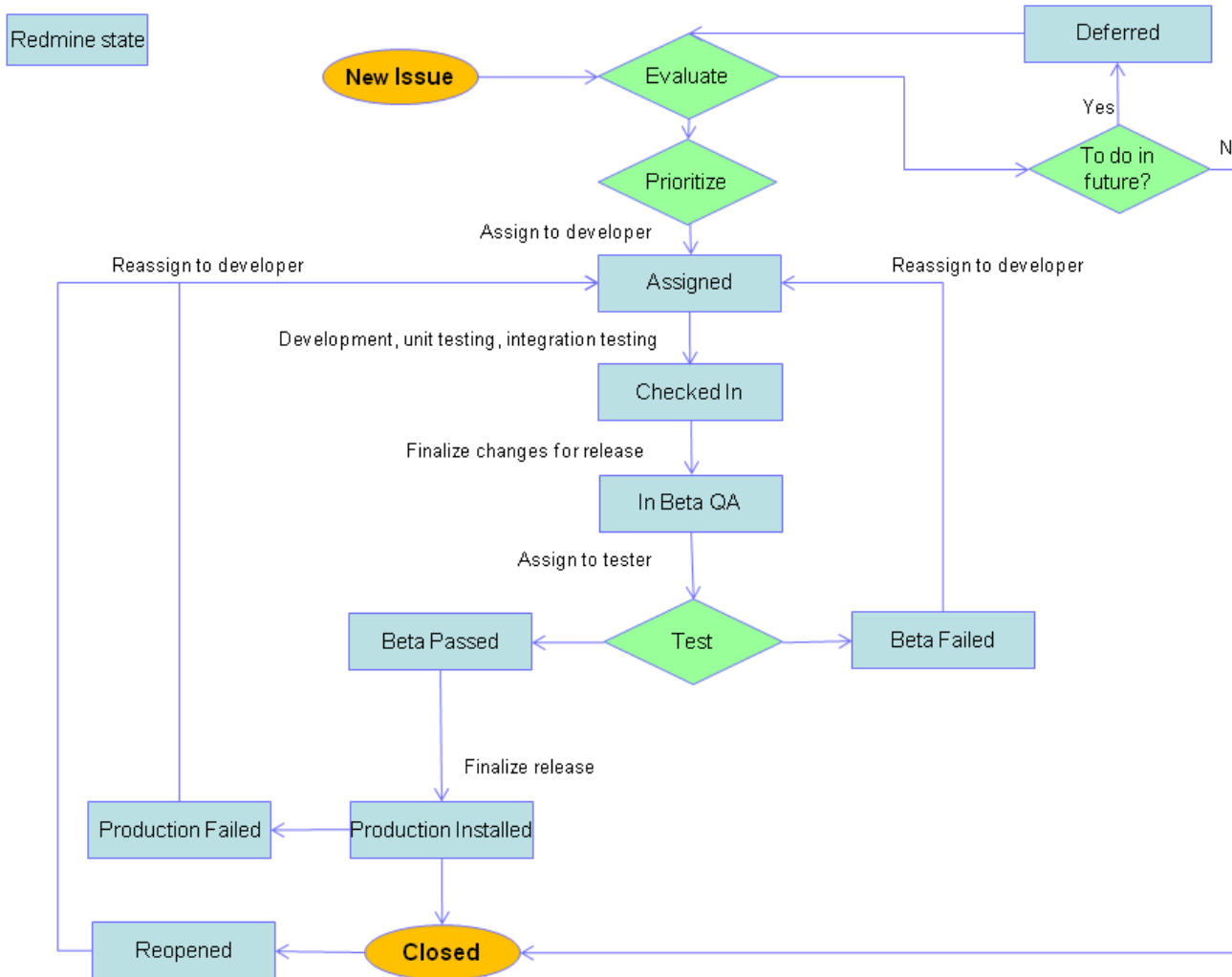
- Program roles:
  - Program Leaders: Review and approve all plans and Statements of Work (SOWs)
  - Principal Investigators (PIs): Develop requirements and SOWs with sponsors
  - Software Development Team Leaders: Oversee all software development, maintenance, and developer testing activities
  - Software Operations Coordinators: Coordinate user requirements and acceptance testing
  - Software Librarian: Organizes software releases, problem reporting and tracking, and regression testing
  - SQA Coordinator: Coordinates SQA plans and procedures
- Tasks evaluated, prioritized and accepted/rejected based on priority, severity, availability of resources, and alignment with sponsor requirements
- NARAC Software Task Management System (STMS)
  - STMS built around the Redmine project management tool (<http://www.redmine.org>)
  - STMS documents requested updates through final resolution
  - STMS tracks all software changes from development to production environments



# Relevant NARAC Program Personnel

- NARAC Program Leader (Sugiyama)
- NARAC Deputy Program Leader (Nasstrom)
- Hardware & Software Systems Development Team Leader (Larsen)
- End User Software Operations Coordinator (Pobanz)
- End User Software Development Team Leader (Eme)
- Central System Operations Coordinator (Foster)
- Central System Software Development Coordinator (Fisher)
- Software Librarian (Weseloh)
- SQA Coordinator (Walker)

# Software Task Management System



## 2. Risk Management

- Evaluation of risks built into the STMS
  - Software changes evaluated for risk by team management
  - Overall software requirements and significant software changes evaluated for risk by Program Management
- Risk evaluation considers:
  - Cost vs. benefit
  - Severity of gap or defect
  - Scope of required software changes
  - Availability of resources
  - Alignment with sponsor and user requirements
- Software reliability continuously monitored and recorded (system performance metrics and statistics)
- Software issues identified and entered in to NARAC Software Task Management System for tracking and resolution





# 3. Configuration Management

- All software components under version control, with labeling and version numbering system
- Mature process for managing software updates, including approvals, acceptance testing, and deployment via STMS
- Maintenance of separate system domains for development and production environments
- Constant monitoring and evaluation of multiple production domains
- Back-ups of all software repositories



# 4. Procurement & Implementation

- Custom software developed and maintained by NARAC personnel
- COTS packages used for standard software applications
- Procurement procedures followed:
  - LLNL vendor procurement review and approval processes
  - NARAC PI review of purchases under their SOWs
  - NARAC TL review and coordinates center hardware purchases
  - NARAC Resource Analyst verification of available budgets
  - NARAC Program Leader approval of all purchases
  - Spreadsheets used to track acquisitions and licenses
- Selection of procured components based on:
  - Support of required capabilities
  - Ease of integration
  - Cost and deployment requirements
  - Vendor reputation and/or past performance

# 5. Requirements Management

- Software requirements are developed from multiple sources
  - Project requirements set by sponsors (documented in SOWs)
  - High-level requirements set by Program Management
  - Corrections and upgrade requests submitted by NARAC staff and external users
  - Performance requirements based on user testing and after action reviews of event support
- Software staff determine software design, scope of changes, and effort requirements
- NARAC PIs, Team Leaders and/or Program Leaders review proposed changes to ensure sponsor, program, and user requirements are met
- NARAC STMS used to manage upgrades and corrections
  - STMS tracks requirements, acceptance testing (production update notes), and deployment
  - Upgrades are new requirements
  - Corrections are weaknesses in meeting a current requirement

# 6. Design & Implementation

- Design & Implementation Approach
  - Formalism tuned to scope of the work
  - Extensive use of patterns and refactoring
  - Effective use of *de facto* standard libraries (e.g. Boost)
  - Effective use of standards and conventions
- Iterative software development
- Software coding conventions
- Formality of review tuned to scope of task
- Version controlled interfaces (i.e. CORBA IDL, WSDL)



# 7. Software Safety

- Not Applicable



# 8. Verification & Validation

- Established test suites with benchmarks
  - Physics models (analytic tests, tracer experiment comparisons)
  - System software components
- Verification performed throughout the development process
  - Developer testing (unit and integration)
  - Tool-based (e.g., JUnit) and custom tests
- Automated regression tests with non-normal inputs (WinRunner)
- Load testing (repeated submission of requests)
- System validation performed by NARAC Operations staff
  - New capabilities
  - Selected regression testing
- In-use testing via system monitoring
  - Hourly automated system checks
  - On-call personnel paging for failure modes

# Load testing

Development: New Client Event

Results Directory :

Iterations :

Type :  ▼

filename	start offset
BigBoom.xml	0::0:01:00
Q2BasicBio.xml	0::0:01:00
Q2BasicChemAgent.xml	0::0:01:00
Q2BasicChemInd.xml	0::0:01:00
Q2BasicMixture.xml	0::0:01:00
Q2BasicOther.xml	0::0:01:00
Q2BasicRad.xml	0::0:01:00
Q2BasicRadHyb.xml	0::0:01:00
Q2BasicTritium.xml	0::0:01:00
Q2ChemSpillMass.xml	0::0:01:00

File Count : 47



# 9. Problem Reporting

- STMS used for problem reporting
  - Assigned roles for managing STMS issues
  - NARAC staff submit problems to STMS directly
  - External customers submit questions and problems to NARAC via phone or email to [narac@llnl.gov](mailto:narac@llnl.gov) (NARAC Customer Support staff enter into STMS)
- Updates and corrections tracked by STMS from submission through development and V&V testing to production deployment



# 10.Training

- NARAC staff training
  - Briefings on new NARAC capabilities
  - Regular NARAC proficiency training and drills tracked by DOE Asset Readiness Management System (ARMS)
  - NARAC User manuals and Web-based documentation
- External user training
  - User Guides and Documentation on the NARAC Web page
  - NARAC Web-based training courses on Emergency Operations Training Academy (EOTA) web-site
  - Web-cast training by NARAC instructors
  - Classroom training courses from NARAC instructors
- Software team training
  - Discipline organization training (SKAs)
  - Mentoring by team leaders and senior software developers

# SQA Guidance

- NARAC SQA activities are intended to be consistent with:
  - DOE O 414.1C – Quality Assurance
  - LLNL ISQAP
  
- As needed, formal documentation follows IEEE templates adapted to meet Program needs, e.g.
  - IEEE 730 – Software Quality Assurance Plans
  - IEEE 828 – Software Configuration Management Plans
  - IEEE 829 – Software Test Documentation

# Lawrence Livermore National Laboratory

Questions?

