# Procedures and Guidelines (PG)

| | | | |
|---|---|---|---|
| **DIRECTIVE NO.** | 320-PG-7120.2.1D | **APPROVED BY Signature:** | ***Original Signed By*** |
| **EFFECTIVE DATE:** | September 4, 2013 | **NAME:** | Susan J. Sekira |
| **EXPIRATION DATE:** | September 4, 2018 | **TITLE:** | Chief, Software Assurance Office |

**COMPLIANCE IS MANDATORY**

**Responsible Office:** Code 320.1/Software Assurance Office

**Title:** Procedure for Planning and Implementing Software Assurance Programs

## PREFACE

### P.1 PURPOSE

This procedure documents Greenbelt's Software Assurance Office (SAO) methodology and implementation of NASA-STD-8739.8, NASA Software Assurance Standard, and NASA-STD-8719.13, NASA Software Safety Standard. Project details regarding specific lifecycle activities can be found in the project's Software Assurance Plan (SAP).

### P.2 APPLICABILITY

This procedure applies to software and systems containing software created and acquired by or for NASA, including Government off-the-shelf (GOTS) software, modified off-the-shelf (MOTS) software, commercial off-the-shelf (COTS) software, and reused software when included in a NASA system. NASA systems include simulators and test software, software supporting Ground Support Electronics (GSE), Facility Control Software Systems, Ground Data Systems, and Flight Software Systems, which includes software running on the main processor, software residing within complex electronics, and software scripts residing in memory.

NOTE: While development and maintenance of Facility Control Software is typically not based on project specific requirements, these systems should be considered when scoping Operational Hazard Analysis. These systems should be analyzed to the extent required to show evidence that the facility operators have implemented sufficient controls related to the potential hazards/failures associated with a particular I&T configuration.

### P.3 AUTHORITY

This procedure adheres to the following NASA standards and requirements for planning and performing software assurance activities.

NASA-STD-8739.8, Software Assurance Standard
NASA-STD-8719.13, Software Safety Standard
NPR 7150.2, NASA Software Engineering Requirements

GPR 7150.4, Software Safety and Software Reliability Process

## P.4 REFERENCES

a. GPR 7150.1, Software Project Process Initiation

b. GPR 7150.2, In-house Software Development and Maintenance

c. GPR 7150.3, Software Acquisition

d. 320-PG-1060.1.1, Chief Safety and Mission Assurance Manager Reporting

e. 320-WI-7120.1.1 Project Mission Assurance Requirements (MAR) Preparation

f. 320-MAR-1001, Standard Mission Assurance Requirements

g. 320-PLAN-1001, Software Assurance (SA) Data Management Plan (DMP)

h. 321-WI-8750.0.1, Software Safety Assessment Process

i. 321-WI-7120.0.3, Hazard Analyses

j. 321-WI-7120.0.4, Hazard Tracking, Resolution & Verification Process

k. IEEE STD 730-2002, IEEE Standard for Software Quality Assurance Plans

l. GSFC-STD-1001, Criteria for Flight and Flight Support System Lifecycle Reviews

m. GSFC Software Assurance Web Site: http://sw-assurance.gsfc.nasa.gov

n. GSFC Software Process Improvement Web site: http://software.gsfc.nasa.gov/

o. Code 320 Handbook: https://320handbook.gsfc.nasa.gov

p. Software Assurance Tracker (SwAT) Instructions on the Software Assurance dedicated server, \\prowler\320\Software Assurance

q. Software Assurance Work Breakdown Structure (WBS) tool, located on the Software Assurance dedicated server, \\prowler\320\Software Assurance

r. Form 4-56, Software Project Initiation Form

## P.5 CANCELLATION

320-PG-7120.2.1C, Procedure for Developing and Implementing Software Quality Programs

## P.6 SAFETY

None

## P.7 TRAINING

Software Assurance practitioners shall have fundamental knowledge in the Safety and Mission Assurance (SMA) Technical Excellence Program (STEP) Software Assurance Competencies through prior experience, training, or certification in methodologies, processes, and standards. While training and/or experience is recommended in all the Software Assurance disciplines, the amount of training may vary at the time of employment or project assignment. Training should be reviewed yearly with the 1st line supervisor to ensure that Software Assurance practitioners have the basic skills to support NASA projects. This includes a detailed knowledge of the NASA Software Assurance Standard, NASA-STD-8739.8, and NASA Software Safety Standard, NASA-STD-8719.13. Training Logs provide a record of completed training, project-specific training, recommended training needs, and required readings and resources.

STEP Level 1 training is required for all Safety and Mission Assurance personnel. The following SATERN and/or additional STEP Level training (or equivalent) is also required for Software Assurance practitioners:

1. NPR 7150.2 Overview
2. Technical Writing and Presentation Skills
3. Basics of Software Assurance for Practitioners
4. Intermediate Software Quality Assurance
5. Software Auditor Skills
6. Software Configuration Management and Data Management
7. Introduction to Software Reliability
8. Overview of Software Safety
9. Software Safety for Practitioners
10. Introduction to CMMI
11. SMA in the Acquisition Process

## P.8 RECORDS

| Record Title | Record Custodian | Retention |
|---|---|---|
| Software Classification and Criticality (GSFC Form 4-56, Software Project Initiation Form) | GSFC Engineering Technical Authority (ETA) for Software with a copy on the Safety & Mission Assurance file server, \\prowler\320\Software Assurance | *NRRS 8/36.5C1 – Handle as permanent. Retention under discussion with NARA. |
| Software Assurance Plan (SAP) | Project's Configuration Management (CM) System with a copy on the Safety & Mission Assurance file server, \\prowler\320\Software Assurance. | |

| | | |
|---|---|---|
| Software Assurance Activity Schedule | The Safety & Mission Assurance file server, \\prowler\320\Software Assurance | |
| Software Assurance Assessment Records (i.e., completed checklists, assessment artifacts, findings, trends, and corrective actions) | Software Assurance Tracker (SwAT) System located at https://swat.gsfc.nasa.gov | |
| Work Breakdown Structure (WBS) | The Safety & Mission Assurance file server, \\prowler\320\Software Assurance | |
| Training Log | The Safety & Mission Assurance file server, \\prowler\320\Software Assurance | |

** NRRS – NASA Records Retention Schedule (NPR 1441.1)*

Additional work products (e.g., weekly and monthly reports) are collected and maintained per 320-PLAN-1001, the Software Assurance (SA) Data Management Plan (DMP).

## P.9 MEASUREMENT/VERIFICATION

Quality control for the Software Assurance function is achieved by a two-prong approach, consisting of weekly communication with the Software Assurance practitioners (e.g., via the Software Assurance Weekly), and tracking/trending of assessment activity and resource levels across GSFC projects, including:

a. Planned vs. Actual Staffing Levels
b. Planned vs. Actual Assessments
c. Number of open vs. closed Assessment findings

Measurement data is captured in the SAE Monthly Reports, as well as individual project dashboards established in the SwAT tool. These measures are used to capture project and program characteristics and trends and drive process improvement.

Audits by external groups are periodically conducted to ensure SAO compliance to NASA Standards, as well as GSFC processes and procedures. Any external audit findings are managed by the sponsoring organization and addressed by the SAO as additional means to improve processes and procedures.

# PROCEDURES

In this document, a requirement is identified by "shall," a good practice by "should," permission by "may" or "can," expectation by "will," and descriptive material by "is."

## 1. OVERVIEW

Software assurance is the planned and systematic set of activities that ensure conformance of software life cycle processes and products to requirements, standards, and procedures.

Software Assurance includes the following disciplines:

a. Software Quality
b. Software Safety
c. Software Reliability
d. Software Verification and Validation
e. Independent Verification and Validation (performed by Code 180). For the purposes of this procedure, IV&V is not covered.

## 2.0 ROLES and RESPONSIBILITIES

Software assurance support to GSFC projects is performed by the Software Assurance Engineer (SAE) and/or Software Safety Engineer (SSE), where one or more persons could perform all the disciplinary functions. Additional subject matter experts (SMEs) in software quality, software safety, and software reliability are available to address discipline-related questions and resolve issues. Software assurance personnel are assigned to projects commensurate with the Basis of Estimate and are managerially and financially independent from the organization developing the software.

Overall management of the GSFC Software Assurance Program is provided by the GSFC Software Assurance Office (SAO) Chief, and project support is managed by the Chief Safety and Mission Assurance Officer (CSO). The roles and responsibilities within the SAO are described below.

### 2.1 Software Assurance Office Chief

Responsibilities include, but are not limited to:

a. Manage the staffing profile and project assignments for the Center's Software Assurance Program
b. Monitor Task Order management of all software assurance tasks for contractor support
c. Prepare the annual Workforce Planning estimates, including travel and training
d. Establish and maintain the interfaces with project management and ensure the working relationship between software assurance practitioners and that of the project and SMA team

e. Collect and analyze customer feedback for process improvement opportunities
f. Champion program improvement initiatives
g. Provide status of office activities and/or issues to SMA management

## 2.2 Software Assurance Lead

Responsibilities include, but are not limited to:

a. Manage cost estimates for software assurance support for current and future projects and/or proposals
b. Negotiate assurance activities, as appropriate
c. Provide technical authority on software classifications when there are consenting opinions
d. Support the Center's Engineering Process Group (EPG) and related activities
e. Provide consultation to SAO Team members on topics related to software assurance and/or software engineering
f. Support process improvement initiatives and collaboration opportunities

## 2.3 Software Assurance Engineer (SAE)

Responsibilities include, but are not limited to:

a. Ensure development and maintenance of the software's classification and criticality
b. Assist in the preparation and maintenance of the project Mission Assurance Requirements (MAR)
c. Ensure the flow down of requirements per the Software Assurance and Software Safety Standards
d. Develop and maintain the project's Software Assurance Plan
e. Generate and maintain a schedule of software assurance activities that cover the life cycle of the project
f. Review software plans and work products
g. Conduct periodic assessments of software processes
h. Perform test witnessing and support activities
i. Support formal/informal reviews
j. Collaborate with Safety, Reliability, and IV&V personnel on software assurance activities
k. Identify/report findings, observations, and risks from software assurance related activities

## 2.4 Software Safety Engineer (SSE)

Responsibilities include, but are not limited to:

a. Review the system safety analyses to ensure that software has been appropriately included
b. Identify software-related hazard causes and software-related hazard controls to mitigate the hazard to an acceptable level of risk
c. Verify that inhibits and controls effectively eliminate or mitigate hazards
d. Participate in reviews concerning safety critical software
e. Provide training and/or consultation on project software safety issues

## 2.5 Software Reliability Engineer (SWRE)

Responsibilities include, but are not limited to:

a. Review reliability analyses to ensure adequate inclusion of software related faults and failures, and software-based mitigations
b. Ensure that software requirements are in synch with the reliability analysis
c. Participate in requirement and code reviews, and test execution for mission critical software
d. Assess impact of discovered defects or proposed changes to mission critical software
e. Trend the maturity of the software from system definition through test

## 2.6 Outreach Manager

Responsibilities include, but are not limited to:

a. Serve as the Software Assurance Liaison with IV&V and Wallops regarding assurance support activities
b. Collaborate on topics of interest across organizations and assurance communities (e.g., the Software Assurance Working Group (SAWG))
c. Support internal and external working groups
d. Maintain information on conferences and workshop offerings, as well as educational opportunities within the GSFC community

See the project-specific Software Assurance Plan for details regarding the assigned Software Assurance personnel and their interactions with key project stakeholders.

# 3.0 PLANNING SOFTWARE ASSURANCE SUPPORT

When planning for the Software Assurance effort on a project, the Software Assurance Engineer (SAE) should have an understanding of the overall mission objectives, mission class, mission concept of operations, preliminary system architecture (if available), the instrument suite (if available), software

classification, and software functionality for the system of interest. In some cases, initial support estimates may need to be made, without details in these areas. However, to the extent possible, the SAE is encouraged to plan for Software Assurance support commensurate with the state of project information available at the time. Later updates to Software Assurance support planning may be made as more information is available or as changes arise.

The major activities involved with Software Assurance Support Planning include performing an initial Software Classification Assessment and Software Safety Criticality Assessment, developing and maintaining a Software Assurance Basis of Estimate, developing the Software Assurance Plan, and developing/maintaining the Software Assurance Activity Schedule.

### 3.1 Initial Software Classification and Software Safety Criticality Assessment

During project formulation, the Software Assurance Engineer and/or Software Assurance Lead shall conduct an independent Classification Assessment of the software. The initial classification assessment identifies and documents the software classification for the software used on the project, including key characteristics. The Software Assurance Engineer's assessment is captured in the Software Project Initiation Form, GSFC 4-56, along with Software Engineering's assessment of the software. The form is maintained by the GSFC Engineering Technical Authority (ETA) for Software and should be reviewed and/or updated at each major milestone review.

The Software Safety Engineer shall identify safety critical software, using the processes defined in GPR 7150.4, Software Safety and Software Reliability Process Document. GSFC Form 4-56 is also used to capture the initial classification and criticality results; however, the software's criticality may not be known until the Preliminary Hazard Analysis (PHA) has been completed.

### 3.2 Software Assurance Basis of Estimate

The Software Assurance Engineer and/or Software Assurance Lead shall develop a Basis of Estimate, which should take into consideration, at a minimum:

a. Statement of Work and scope
b. Software Assurance (SA) Work Breakdown Structure (WBS)
c. Effort estimates with supporting assumptions, e.g., software assurance effort commensurate with software size
d. Supporting cost model estimates and available analogies, e.g., comparison of software assurance effort on previous projects, cost model results from software engineering, predictions of effort increases due to risk assumptions and uncertainty
e. Project Schedule

f. Procurements
g. Acquisition approach (if applicable)
h. Significant cost and risk drivers
i. Risk items, issues, and/or any known liens

Using the SA WBS tool and available information from above, Software Assurance will estimate the specific tasks to be performed and the associated effort estimate. This Basis of Estimate will be reviewed by the SAO and can be discussed with the Project CSO and/or Project Management and used as a means for negotiating an increase or decrease in the scope of work. Software Assurance personnel should have a clear understanding of the Software Assurance requirements and identify the level of effort associated with each project support activity. The Software Assurance Basis of Estimate may need to be updated as the software and system progresses through the lifecycle. The Basis of Estimate should be generated with particular consideration for the overall mission objectives, the system concept of operations, software functionality, software classification and safety criticality, project risk posture, software and software assurance assumptions, and uncertainties.

### 3.3. Project-specific Software Assurance Plan

The Software Assurance Engineer is responsible for developing and maintaining the Software Assurance Plan (SAP) using the latest SAO template and results from the WBS. The SAP serves as a tool for communicating the expectation, scope, and accountability of the Software Assurance effort. The SAP details the documentation, reviews, analyses, and assessments required to accomplish software assurance and may be organized as necessary with respect to the particular project of interest. Note that the SAP, coupled with this PG, provides a comprehensive overview of the software assurance activities and methodology.

The SAP should be reviewed by the SAO and approved by the CSO and the project.

### 3.4 Software Assurance Activity Schedule

The Software Assurance Engineer shall develop and maintain a schedule of activities commensurate with the WBS and the Triggers Table (Appendix C). The activity schedule should include the tasks to be performed (product or process assessments), an estimation of the percentage of the activity that has been completed, planned start date, planned end date, actual start date, actual end date, and any comments or rationale associated with that task. At a minimum, schedules should be updated monthly. Information regarding planned peer reviews, milestone reviews, test activities, and/or standing meetings with key stakeholders should be captured in the project's schedule and made available to the SAE.

## 4.0 IMPLEMENTING SOFTWARE ASSURANCE

At NASA, Software Assurance is accomplished through the roles and responsibilities of two functional groups: the Acquirers and the Providers. The Provider designs, develops, implements, tests, operates, and maintains software products/systems, while the Acquirer performs oversight of Provider activities and ultimately accepts the software from the Provider organization. Each organization providing support may have different responsibilities and/or focus areas. The activities below provide a high-level summary of what software assurance entails, with details captured in the project's Software Assurance Plan (SAP). Additional guidance can be found in the WBS Tool and the Code 320 Handbook.

Key activities that are specific to the Provider and/or the Acquirer are highlighted below.

### 4.1 Process and Product Quality Assurance (PPQA)

Software Assurance is responsible for providing an independent and objective assessment of the processes and associated work products against the project's applicable process descriptions, standards, and procedures. The Software Assurance Engineer (SAE) shall work with the software project to identify the appropriate processes and products to fit the project's needs. Throughout the project lifecycle, the SAE will review and evaluate the project's compliance to their documented plans and procedures, and document, track, and communicate any noncompliances with the project's stakeholders. Assessment checklists are available for use on the Prowler site and should be tailored to the provider's processes and procedures.

The primary focus of Acquirer Software Assurance is one of oversight, where SAEs are responsible for ensuring the flow down of assurance requirements and the successful implementation of Provider Software Assurance. SAE's should conduct periodic assessments of the Provider's software assurance efforts and compliance to their documented Software Assurance Plan (SAP) and supporting internal policies, procedures, etc. Assessments can occur during site visits or by reviewing applicable Provider software assurance records/reports, participation in reviews and Provider status tag-ups, and/or regular communication with the Provider. In the event the SAE identifies potential risks in the Provider assurance program, Acquirer Software Assurance support may be expanded to include full-up independent assessments of provider software processes.

Provider Software Assurance has the responsibility of ensuring that the software provider complies with their Software Development Plan/Software Management Plan, and associated processes and procedures. This includes assessments of all the CMMI Level 2 process areas, as well as risk management, where assessments are driven by the project delivery of products.

See Appendix C, Assessment Trigger Table, for the required product and process assessments, and the recommended assessment frequency.

### 4.2 Software Safety and Software Reliability

Software Safety and Software Reliability are disciplines that focus on assuring that software associated hazardous functions or mission critical functions are appropriately addressed.

For software safety, such efforts include, but are not limited to 1) identifying, analyzing and tracking software-related hazards, conditions and events along with the associated controls, mitigations and inhibits within a flight or ground system throughout the entire software and system development life cycle, and 2) verification and validation, through testing and/or analysis, the correct and safe software operations, including the controls and mitigations.

For software reliability, such efforts include, but are not limited to 1) identifying and tracking failure modes with associated impacts and mitigations for all mission critical software within a flight or ground system throughout the entire software and system development life cycle, and 2) verification and validation of the proper implementation of mission critical software through testing and/or analysis.

Both software safety and software reliability efforts require a close collaboration with project safety and reliability engineering, as well as the system/software engineering. Detailed software safety and reliability requirements and responsibilities can be found in NASA STD 8739.8, Software Assurance Standard, NASA STD 8719.13C, Software Safety Standard, NASA NPR 7150.2A, Software Engineering Requirements, and GPR 7150.4, Software Safety and Reliability Process.

### 4.3 Test Activities

Software Assurance has the responsibility to observe/monitor the formal and acceptance software testing process from the pre-test, testing, and post-test phases to verify satisfactory completion and outcome. This entails assuring that all tests, including pre / post-test and dry runs, are conducted using approved test procedures and appropriate test tools, version controls, and documented CM processes, and that test anomalies are identified, documented, addressed, and tracked to closure. In addition, SA will assure that assumptions, constraints, and test results are accurately recorded to substantiate the requirements verification/validation status. This may include, but is not limited to, audits of test documentation and attendance of testing reviews. In those instances when it may not be realistic to attend all formal and acceptance software testing events in person, Software Assurance may monitor selected test activities and review/audit the results of the testing event for satisfactory completion and outcome. Similarly, test witnessing may be focused on the criticality of the function being tested or the fidelity of the test bed.

When GSFC is the Acquirer of software, Software Assurance may physically witness all tests or a subset of the tests, review just the test results from the Provider, or delegate test witnessing activities to another assurance agent (e.g., DCMA). GSFC Software Assurance should work with the Provider Software Assurance to ensure the same or adequate software assurance oversight during testing activities.

## 4.4 Formal/Informal Reviews

It is incumbent upon Software Assurance to participate in informal reviews (e.g., Software and Project-level monthly or quarterly status reviews), milestone reviews, and engineering peer reviews.

### 4.4.1 Milestone Reviews

In support of key lifecycle reviews, the SAE should consult GSFC-STD-1001, Criteria for Flight and Flight Support System Lifecycle Reviews, for guidance on what should be presented and work with the project to reach a consensus on the entrance/exit review criteria. For Mission-level reviews, the Mission review criteria should be applied.

Using the agreed upon review criteria, the Provider Software Assurance organization is responsible for preparing the assurance information to be reviewed at Software Milestone Reviews and Project Level Milestone Reviews. The SAE's input should focus on the status of the software engineering plans and compliance matrices, review and approval of all formal software deliverables, assessments for all relevant software engineering processes, and planned assurance activities in the upcoming phase.

The Acquirer Software Assurance organization is responsible for reviewing the information provided at Software Milestone Reviews and Project Level Milestone Reviews against the criteria, expectation, and intent of the Software Assurance requirements. The Acquirer SAE also provides input that focuses on summarizing the Process and Product Quality Assurance (PPQA) activity, including the status of external PPQA assessments, review and approval of software assurance plans and compliance matrices, and any additional on-site or off-site monitoring activities.

In the event of Requests for Action (RFAs), Software Assurance should work directly with the originators and owners of the software-related RFAs to confirm the nature of the concern and to assure that a proper response is provided, and verify that the action is worked to closure.

### 4.4.2 Peer Reviews

In addition to providing oversight of the process for formal and informal software reviews, SAE's should participate as active members in the peer review process for software requirements, design specifications, code, and test products/artifacts. This is the most effective means of verifying requirement traceability and the application of best practices related to the design and test of critical systems, and ensuring that faults are not introduced into the system by way of process non-compliance.

## 4.5 Problem Reporting and Corrective Action

Software Assurance will generate, track, and trend assessment findings and observations in the GSFC Software Assurance Tracker (SwAT) tool. Results will be communicated to the project and the SMA team through weekly and monthly reporting vehicles, as well as regular tag-ups.

## 4.6 Risk Management

Software assurance assists in risk mitigation by minimizing defects, preventing problems and enabling improvement of software products and services. SA personnel will participate in Risk Management meetings as directed by the CSO and will report any software risks to the CSO and the project's Risk Manager, as necessary.

## 4.7 Cross-Cutting Support Activities

Throughout the lifecycle of the project, Software Assurance supports numerous status meetings and control boards. Depending on the project, these could include participation on Configuration Control Boards (CCBs) and Engineering Review Boards (ERBs), support at Monthly Project Status Reviews, and scheduled meetings with the CSO, SMA Team, and/or software development team (s). These meetings and boards provide the SAE the opportunity to gain insight into those areas that most concern the project and where software may have an impact to the project's critical path. When software is being acquired, the SAE should also support regular tag-ups and reviews with the Provider's Software Assurance Organization.

The SAE is responsible for providing a Weekly and Monthly Report to their CSO and other CSO-selected stakeholders throughout the lifecycle. This reporting chain is independent of the provider's project management and software development functions. Reporting guidance can be found in the 320 Handbook.

## 4.8 Assurance Tools, Techniques, and Methodologies

GSFC SA personnel utilize assets provided by multiple organizations that support the software assurance function. These assets range from CM approved requirements/standards, and job aids (such as checklist, and templates), to issue trackers, and dedicated servers. The following table lists the supporting organizations and describes tools and resources that are available:

| | |
|---|---|
| The Goddard Space Flight Center (GSFC) Software Assurance Tracker (SwAT)<br><br>http://swat.gsfc.nasa.gov | Issue tracking system used by GSFC Software Assurance Engineers. The SwAT serves as the repository for software assurance artifacts and work products (i.e., schedules, assessments, findings, and observations). Metrics are available via the Dashboard function |
| The GSFC Software Assurance Website<br><br>http://sw-assurance.gsfc.nasa.gov | GSFC Software Assurance Website provides information regarding software assurance support services, roles and responsibilities, and links to additional guidance and tools |
| GSFC Software Process Improvement Website<br><br>http://software.gsfc.nasa.gov | Hosted by the GSFC Engineering Process Group (EPG), the SPI website provides documented processes, procedures, guidelines, templates, checklists, forms, tools, and other assets that describe and/or support GSFC-approved methods for managing and developing software |
| NASA Online Directives Information System (NODIS)<br>http://nodis.gsfc.nasa.gov | Repository for NASA Agency-level directives and requirements |
| Goddard Directives Management System (GDMS)<br>http://gdms.gsfc.nasa.gov | Repository for GSFC policy and implementing instructions (policies, guidelines, requirements, and work instructions) |
| System for Administration, Training, and Educational Resources for NASA (SATERN)<br><br>https://satern.nasa.gov | Online training portal for NASA |
| The NASA Standards And Technical Assistance Resource Tool (START)<br><br>https://standards.nasa.gov | Repository for NASA Technical Standards |
| Software Assurance Server<br>\\prowler\320\Software Assurance | Internal Server used to maintain SA work products and templates |

## Appendix A – Definitions

**A.1** **Acquirer -** The entity or individual who specifies the requirements and accepts the resulting software products. The acquirer is usually NASA or an organization within the Agency, but can also refer to the Prime contractor – subcontractor relationship as well.

**A.2** **Assessment –** An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria. [IEEE 610.12]

**A.3** **Chief Safety and Mission Assurance Officer (CSO) –** Code 300 personnel responsible for supporting the Product Development Lead (PDL) in the coordination of the definition and implementation of a Project Systems Safety and Mission Assurance Program (SSMAP).

**A.4** **Configuration Management (CM) –** A discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements. [IEEE 610.12]

**A.5** **Finding –** Non-compliance to a requirement, procedure, standard, or specification.

**A.6** **Integrated Independent Review (IIR) –** One of a series of system-level reviews conducted at critical project/product milestones in accordance with GPR 8700.4. IIRs build upon the results of a robust set of engineering peer reviews. IIR examples include System Concept review (SCR), Critical Design Review (CDR), and Mission Operations Review (MOR).

**A.7** **Mission Critical Software -** Software that can cause, contribute to, or mitigate the loss of capabilities that are essential to the primary mission objectives. The software reliability assessment and analysis is focused on failure modes specific to post-separation mission phases. Mission-critical software is identified based on the results of the Failure Modes and Effects Analyses (FMEA) and the Probabilistic Risk Assessment (PRA). Examples of mission-critical software can be found in all types of systems, including Flight, Ground Support System, Mission Operations Support Systems, and Test Facilities.

**A.8** **Observation -** A statement of fact (positive or negative) based on objective evidence.

**A.9** **Process Assessment –** A systematic examination to determine whether a software process is being performed in accordance with documented plans, procedures, etc.

**A.10** **Product Assessment –** A systematic examination to determine whether a software product meets specified requirements and standards.

**A.11** **Product Development Lead (PDL) –** The manager or leader with overall responsibility for managing the design activity, managing the technical and organizational interfaces identified during design planning, and where required, forming and leading the Product Development Team (PDT). The term refers to flight project managers, principal investigators, mission managers, instrument managers, software managers, lead engineers, etc.

**A.12** **Provider –** The entities or individuals, who design, develop, implement, test, operate, and maintain the software products. A provider may be a contractor, a university, a separate organization within NASA or within the same organization as the acquirer.

**A.13** **Safety Critical Software** – Software that can cause, contributes to, or mitigates human safety hazards or damage to flight hardware and facilities. The software safety assessment and analysis is focused on hazards specific to Integration and Test, launch, and up through spacecraft separation from the launch vehicle (except for International Space Station (ISS) payloads that have constant human presence) and re-entry/recovery (where applicable). Safety-critical software is identified based on the results of the hazard analysis and the results of the Orbital Debris Assessment Report/End-Of-Mission Plan (where applicable). Examples of safety-critical software can be found in all types of systems, including Flight, Ground Support System, Mission Operations Support Systems, and Test Facilities.

**A.14** **Software -** Computer programs, procedures, scripts, rules, and associated documentation and data pertaining to the development and operation of a computer system. Software includes programs and data. This also includes COTS, GOTS, MOTS, reused software, auto generated code, embedded software, and open source software components.

**A.15** **Software Assurance (SA)** – The planned and systematic set of activities that ensure that software life cycle processes and products conform to requirements, standards, and procedures. [IEEE 610.12] For NASA this includes the disciplines of Software Quality (SQ), Software Safety, Software Reliability, Software Verification and Validation (V&V), and Independent Verification and Validation (IV&V).

**A.16** **Software Quality (SQ)** – The discipline of software quality is a planned and systematic set of activities to ensure quality is built into the software. It consists of software quality assurance, software quality control, and software quality engineering.

**A.17** **Software Quality Assurance (SQA) -** The function of software quality that assures that the standards, processes, and procedures are appropriate for the project and are correctly implemented.

**A.18** **Software Quality Control** – The function of software quality that checks that the project follows its standards, processes, and procedures, and that the project produces the required internal and external (deliverable) products.

**A.19** **Software Quality Engineering** – The function of software quality that assures that quality is built into the software by performing analyses, trade studies, and investigations on the requirements, design, code, and verification processes and results to assure that reliability, maintainability, and other quality factors are met.

**A.20** **Software Quality Personnel -** Personnel responsible for providing SQ support to the CSO. This includes software quality engineers, Defense Contract Management Agency (DCMA) specialists, or support provided under the NASA Contract Assurance Services (NCAS). Note: The CSO may also perform the duties of a software quality person.

**A.21** **Software Reliability** – The discipline of software assurance that (1) defines the requirements for software controlled system fault/failure detection, isolation, and recovery; (2) reviews the software development processes and products for software error prevention and/or reduced functionality states; and (3) defines the process for measuring and analyzing defects and defines/derives the reliability and maintainability factors.

**A.22** **Software Requirements Traceability Matrix (SRTM)** – A tool developed and maintained by software engineering that traces software requirements back to system requirements and forward to design, code, and test.

**A.23** **Validation** – Confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are correct and fulfilled (Did we build the right thing?).

**A.24** **Verification** - Confirmation by examination and provision of objective evidence that specified requirements have been fulfilled (Did we build it right?).

## Appendix B – Acronyms

| | |
|---|---|
| **CCB** | Configuration Control Board |
| **CDR** | Critical Design Review |
| **CM** | Configuration Management |
| **CMMI** | Capability Maturity Model Integrated |
| **COTS** | Commercial Off-the-Shelf |
| **CSO** | Chief Safety and Mission Assurance Officer |
| **DCMA** | Defense Contract Management Agency |
| **DMP** | Data Management Plan |
| **EPG** | Engineering Process Group |
| **ERB** | Engineering Review Board |
| **ETA** | Engineering Technical Authority |
| **FMEA** | Failure Modes and Effects Analyses |
| **FY** | Fiscal Year |
| **GDMS** | GSFC Directives Management System |
| **GOTS** | Government Off-the-Shelf |
| **GPR** | Goddard Policy Requirements |
| **GSFC** | Goddard Space Flight Center |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **IIR** | Integrated Independent Review |
| **ISS** | International Space System |
| **IV&V** | Independent Verification and Validation |
| **MAR** | Mission Assurance Requirements |
| **MOR** | Mission Operations Review |
| **MOTS** | Modified Off-the-Shelf |
| **NA** | Not applicable |
| **NASA** | National Aeronautics and Space Administration |
| **NCAS** | NASA Contract Assurance Services |
| **NODIS** | NASA Online Directives Information System |
| **NRRS** | NASA Record Retention Schedules |
| **NPR** | NASA Procedural Requirements |
| **PDL** | Product Development Lead |
| **PDR** | Preliminary Design Review |
| **PDT** | Product Development Team |
| **PG** | Procedures and Guidelines |
| **PHA** | Preliminary Hazard Analysis |
| **PPQA** | Process and Product Quality Assurance |
| **PRA** | Probabilistic Risk Assessment |
| **RFA** | Request for Action |
| **SA** | Software Assurance |

| | |
|---|---|
| **SAE** | Software Assurance Engineer |
| **SAO** | Software Assurance Office |
| **SAP** | Software Assurance Plan |
| **SATERN** | System for Administration, Training, and Educational Resources for NASA |
| **SAWG** | Software Assurance Working Group |
| **SCM** | Software Configuration Management |
| **SCR** | System Concept Review |
| **SMA** | Safety and Mission Assurance |
| **SMP/PP** | Software Management Plan/Product Plan |
| **SOW** | Statement of Work |
| **SQ** | Software Quality |
| **SQA** | Software Quality Assurance |
| **SRTM** | Software Requirements Traceability Matrix |
| **SSE** | Software Safety Engineer |
| **SSMAPP** | Systems Safety and Mission Assurance Plan |
| **STD** | Standard |
| **STEP** | SMA Technical Excellence Program |
| **SwAT** | Software Assurance Tracker |
| **SWRE** | Software Reliability Engineer |
| **V&V** | Verification and Validation |
| **WBS** | Work Breakdown Structure |
| **WI** | Work Instruction |

# Appendix C – Assessment Triggers

## In-House Provider Assessments

For In-house development efforts, the Software Assurance Engineer (SAE) is responsible for ensuring that the Software Provider complies with their documented Software Development Plan/Software Management Plan, and associated processes and procedures. This includes assessments commensurate with the software's classification, where the scheduling of assessments is driven by the project delivery of products, as well as the life cycle phase. The SAE should consult the project's schedule for software deliverables and milestone dates.

## Acquisition Assessments

As a member of the Acquisition Team, SAE's are required to conduct periodic assessments of the Provider's software assurance efforts and compliance to their documented Software Assurance Plan (SAP). This comprehensive assessment assures that the Provider Software Assurance Team is monitoring all the software development processes and products and maintaining records of their assessments. It also provides the SAE insight into potential risks in the provider assurance program, and as an escalation, may expand Acquirer support to include comprehensive independent assessments of provider software processes.

The SAE also has the responsibility of ensuring that the in-house Acquisition Team is following their documented Software Acquisition Management Plan (SAMP). This entails reviewing their plan, associated procedures, Acquisition Schedule, and monthly Status Reports.

## Assessments Triggers

The table below captures the required process and product assessments to be conducted, and the following additional detail:

- Whether the assessments should be conducted for GSFC developed (internal) systems or acquired (external) software systems
- Where the artifacts of the assessments should be maintained*
- Recommended triggers and frequencies for conducting the assessment**

For each software process, the table also identifies the work products that are subject to documented stand-alone assessments, and work products that are required to be reviewed only in conjunction with the software process assessment (identified as NA in the Artifacts column).

*All process assessment artifacts and only a subset of the product assessments need to be documented in Software Assurance Tracker (SwAT). In most instances, artifacts from software product assessments

that are conducted in concert with the project or software team's peer review cycle will be maintained in the corresponding (CM) tool, such as the project's MIS, or the software team's MKS instance.

**Where assessments are triggered by milestone reviews, this is a reference to the software system-level (i.e., Spacecraft, Instrument, or Ground System) reviews, specifically SRR, PDR, CDR, TRR, SIR, and ORR. If lifecycle phases span multiple years, in years where there is no milestone review, then process assessments should be annually, and scheduled such that the results can be used as input into FY resource planning.

| Processes and Associated Products | Applicability | Artifacts | Triggers/Frequency |
|---|---|---|---|
| **Project Planning** [SWE-032] | I | S | **Process Frequency:** 60 Days Prior to software milestone reviews, as a result of re-baselining, or annually should the time between reviews span multiple years<br><br>**Product assessments** (e.g., plans, forms) per the project's planned delivery schedule |
| Software Development/Management Plan [SWE-013] | B | S | |
| Software CM Plan [SWE-013] | B | S | |
| Software Risk Management Plan [SWE-032] | B | S | |
| Software Safety Plan [SWE-130] | B | S | |
| Software Assurance Plan [SWE-13] | E | S | |
| Software Test Plan [SWE-013, SWE-065] | B | S | |
| Software Maintenance Plan [SWE-013] | B | S | |
| Software Inventory & Classification [SWE-020] | B | P | |
| Software Cost Estimate [SWE-015] | I | NA | |
| Software Schedule [SWE-016] | I | NA | |
| MAR,PAIP/MAIP | B | NA | |
| **Project Monitoring and Control** [SWE-032] (includes Measurement and Risk Management) | I | S | **Process Frequency:** 60 Days Prior to software milestone reviews, or annually should the time between reviews span multiple years<br><br>**BSR product assessments** every 6 months<br><br>**Milestone review packages** assessed upon receipt and in advance of the review |
| Branch Status Review (BSR) Packages & Minutes | I | S | |
| Milestone Review Packages (e.g., SRR, PDR, CDR, TRR, SWAR, SIR, ORR) | B | P | |
| Metrics Database/Tool | I | NA | |
| Risk Database/Tool | I | NA | |
| Risk Board Minutes | I | NA | |
| Action Item Database/Tool | I | NA | |
| RFA Database/Tool | I | NA | |
| Internal Review Board (IRB) Minutes | I | NA | |

| Processes and Associated Products | Applicability | Artifacts | Triggers/Frequency |
|---|---|---|---|
| **Configuration Management** [SWE-032] | I | S | **Process Frequency:** 60 Days Prior to software milestone reviews, or annually should the time between reviews span multiple years<br><br>**DML product assessments** every 6 months<br><br>**VDD product assessments** per formal or external software delivery |
| Data Management List (DML) and/or Configured Articles List (CAL) | I | S | |
| Configuration Status Reports [SWE-083] | I | NA | |
| Software Version Description Document [SWE-063, SWE-077] | B* | S | |
| Software Change Request [SWE-080] | B* | P | |
| Software Problem Reports [SWE-080] | B* | P | |
| FCA/PCA Artifacts [SWE-084] | B* | P | |
| CCB Minutes/Artifacts | I | NA | |
| CM database/tool | I | NA | |
| **Requirements Management** [SWE-032] | I | S | **Process Frequency:** 60 Days Prior to software milestone reviews, or annually should the time between reviews span multiple years<br><br>**Product assessments** of the requirements and the requirements traceability matrix per every major build or release |
| Software Requirements Specification [SWE-049] | B | S | |
| Interface Design Description [SWE-056] | B | S | |
| Requirement Traceability Matrix [SWE-030], [SWE-052], [SWE-068], [SWE-104] | I | NA | |
| Command and Telemetry Dictionary [SWE-049] | I | P | |
| Requirements Verification Criteria [SWE-028] | I | NA | |
| Requirements Peer Review Artifacts [SWE-087,SWE-088, SWE-089] | I | P | |
| Requirements Management Tool/Database | I | NA | |

| Processes and Associated Products | Applicability | Artifacts | Triggers/Frequency |
|---|---|---|---|
| **Verification and Validation** [SWE-028 – SWE-031] | I | S | **Process Frequency:** In conjunction with Software TRR, SIR, and Flight I&T Readiness Reviews, and ORR, or annually should the time between the reviews span multiple years.<br><br>**Product assessments** of the test plans per the project's planned delivery |
| Design and Code V&V Criteria [SWE-028] | I | NA | |
| PDR,CDR Artifacts [SWE-56],SWE-087,SWE-088, SWE-089] | B | P | |
| Code Review Artifacts [SWE-087,SWE-088, SWE-089] | I | NA | |
| Build Test Procedures [SWE-065] | I | P | |
| Build Software Test Report [SWE-065] | I | P | |
| Test Readiness Review Package/Artifacts | B | P | |
| Acceptance Test Procedures [SWE-065] | B | P | |
| Acceptance Test Reports [SWE-065] | B | P | |
| Acceptance Review Package/Artifacts | B | P | |
| Requirements Verification Matrix [SWE-030], [SWE-068], [SWE-104] | I | NA | |
| Development Folder/Environment | I | NA | |

| Processes and Associated Products | Applicability | Artifacts | Triggers/Frequency |
|---|---|---|---|
| **Software Assurance /PPQA** [SWE-032], [SWE-022] | E | S | **Process Frequency:** First assessment prior to SRR and at least annually thereafter during a site visit<br><br>**Product assessments** upon delivery (per the contractual agreement) and with any subsequent updates. |
| Supplier Software Assurance Plan (SAP) and compliance matrices [6.3.1] | E | S | |
| Software Safety Plan | E | S | |
| Software Reliability Plan | E | S | |
| Software Assurance procedures | E | NA | |
| Software Assurance Assessment Schedule | E | NA | |
| Software Assurance Assessment Criteria | E | NA | |
| Software Assurance Assessment Records [6.6.1] | E | NA | |
| Software Assurance Status Reports | E | NA | |
| **Supplier Agreement Management** [SWE-032],[SWE-034],[SWE-036],[SWE-037],[SWE-038] | I | S | **Process Frequency:** 60 Days Prior to software milestone reviews, or annually should the time between reviews span multiple years.<br><br>**SAMP assessments** per the project's planned delivery<br><br>**ASR and Schedule product assessments** every 6 months with a focus on metrics and risk |
| Software Acquisition Management Plan (SAMP) | I | **S** | |
| Contractual Documents (SOW, MOU,MOA) | I | NA | |
| Subcontract Agreements | I | NA | |
| Acquisition Management Team (AMT) Schedule | I | S | |
| Acquisition Status Reports | I | S | |

**Key:**
Applicability

- Internal (I) – Indicates that the assessment is applicable for internal software development efforts
- External (E) – Indicates that the assessment is applicable for external software development efforts

- Both (B) – Indicates that the assessment is applicable for both internal and external development efforts
- (B*) – The assessment is applicable for external efforts in conjunction with or post software acceptance Artifacts
- SwAT (S) – Indicates that records from the assessment are maintained in the SwAT tool
- Project (P) – Indicates that records from the assessment are maintained in a project system
- NA – The assessment is performed as part of the process assessment and the artifacts are maintained accordingly

## CHANGE HISTORY LOG

| Revision | Effective Date | Description of Changes |
|---|---|---|
| Baseline | 11/17/2003 | Initial Release |
| A | 7/27/2004 | Modified title of PG, added configuration control number to the NASA Software Policies NPD, added 3 new references to P.5, modified recommended training in P.7, modified definitions for "finding" and "observation", added an SQ Reporting Form to P.8, deleted option to include the SAP in the Project's Surveillance Plan, provided updates to Table 4.0-1, added references to the Software Quality Assessment Process 303-WI-7120.1.1 in Sections 4.2 and 4.3, deleted the Software Safety Plan as a product in Table 4.2-1, and updated naming convention to several software quality work instructions in Tables 4.2-1 and 4.3-1. |
| B | 1/25/2010 | Removed non-requirements and revised to add clarity regarding requirements. |
| B | 03/13/2010 | Re-issued to assign new directive number to reflect new Code 320 organization. Administratively extended for 1 year from original expiration date. |
| C | 03/30/2010 | Provided updates to the document's Reference Section, including deletion of two Work Instructions, revised the Measurement Section, added training requirements, updated Table 4.0-1, added the software classification and assessment requirements, made necessary nomenclature updates to align with the current SMA organization, and added Appendix C. |
| D | 09/04/2013 | Modified title. Added new links and tools to the Reference Section, updated the Training section to include STEP, added new records to P.8, updated roles and responsibilities in Section 2.0, reformatted Sections 3 and 4, included a new section on Basis of Estimates, added text to differentiate between Acquirer and Provider Software Assurance, and updated the Triggers Table in Appendix C. |