**COMPLIANCE IS MANDATORY**



**Responsible Office: Q/Safety and Mission Assurance**

# Subject: GRC Software Assurance and Software Safety

## TABLE OF CONTENTS

**Appendix B: Acronyms**

**Change History Log**

**Distribution: BMS Library**

# PREFACE

## P.1 PURPOSE

a. This Glenn Procedural Requirement (GLPR) establishes the Glenn Research Center (which includes John H. Glenn Research Center at Lewis Field and Neil A. Armstrong Test Facility and hereby referred to as GRC) Software Assurance and Software Safety (SASS) requirements as derived from NASA-STD-8739.8, NASA Software Assurance and Software Safety Standard.

b. The GLPR defines a systematic approach to Software Assurance (SA), software safety, and Independent Verification and Validation (IV&V) for software created, acquired, provided, or maintained by or for GRC.

c. The objectives of this GLPR include:

(1) Establish GRC level procedural requirements for Software Assurance and Software Safety requirements used to produce and sustain software.

(2) Ensuring that the software systems are safe and that the software safety-critical requirements and processes are followed for GRC.

(3) Ensuring that the software systems are secure for GRC.

## P.2 APPLICABILITY

a. This directive is applicable to all organizations at GRC, including contractors, grant recipients, and other parties to the extent specified in the associated contracts, grants, or agreements. This statement alone is not sufficient to stipulate requirements for the contractor, grant recipient, or agreement. NASA-STD-8739.8 provides requirements for NASA contracts, grant recipients, or agreements to the responsible NASA project managers and contracting officers that are made mandatory through contract clauses, specifications, or statements of work (SOWs) in conformance with the NASA Federal Acquisition Regulation (FAR) Supplement or by stipulating in the contracts, grants, or agreements which of the NASA Procedural Requirement (NPR) or NASA Standard requirements apply.

b. This directive is applicable to documents developed or revised after the effective date of this GLPR.

c. All mandatory actions (i.e., requirements) are denoted by statements containing the term “shall.” The terms “may” denotes a discretionary privilege or permission, “can” denotes statements of possibility or capability, “should” denotes a good practice and is recommended, but not required, “will” denotes expected outcome, and “are/is” denotes descriptive material.

d. All document citations are assumed to be the latest version, unless otherwise noted.

e. This directive applies to software acquisition, development, maintenance, operations, management, retirement, and assurance activities as defined applicable in NPR 7150.2 and NASA-STD-8739.8.

f. The SASS efforts performed by GRC for other NASA Centers will use the policies and procedures of

the lead center. If the lead center does not require specific policies or procedures, this policy will be utilized or tailored as deemed appropriate, provided the tailoring is approved by the GRC Software Assurance Technical Authority (TA). If the lead center policies or procedures conflict with those of GRC, the lead center policies will take precedence.

g. This GLPR applies to the assurance of software and systems containing software created by or for NASA projects, programs, facilities, and activities and defines the requirements for those activities. This includes NASA system software residing in non-volatile memory or volatile system memory, software included in Ground Support Equipment (GSE), Flight Software Systems, simulators, test software, software supporting test facilities, software supporting units under test, software supporting data acquisition systems, ground data system software, and facility control software systems.

h. All GRC Software Assurance and Safety and Mission Assurance (SMA) Technical Authorities shall execute "Technical Authorities" duties and requirements as specified in NPR 7150.2, NASA-STD-8739.8, and as delegated in Glenn Plan (GLP) 1120.1, NASA John H. Glenn Research Center Technical Authority Implementation Plan.

i. In this document, the phrase “Software Assurance and Software Safety Tasks” means that the roles and responsibilities for completing these requirements may be delegated within the project consistent with the scope and scale of the project.

j. This GLPR applies to GRC Program/Project/Facility (P/P/F) software and systems that contain software.

k. This GLPR applies to organizations in their roles as both acquirers and providers.

## P.3 AUTHORITY

a. NASA Policy Directive (NPD) 7120.4, NASA Engineering and Program/Project Management Policy

b. NASA Procedural Requirements (NPR) 7150.2, NASA Software Engineering Requirements

c. Glenn Policy Directive (GLPD) 7150.2, GRC Software Engineering Requirements

c. NASA-STD-8739.8, NASA Software Assurance and Software Safety Standard

## P.4 APPLICABLE DOCUMENTS AND FORMS

a. NPR 7120.5, NASA Space Flight Program and Project Management Requirements

b. NPR 8715.3, NASA General Safety Program Requirements

g. GLPR 7120.5.30, Space Assurance Requirements

h. GLP 1120.1, NASA John H. Glenn Research Center Technical Authority Implementation Plan

h. GRC 1707 Form, Special Approvals and Affirmations of Requisitions

## P.5 MEASUREMENT/VERIFICATION

Not Applicable

## P.6 CANCELLATION

This GLPR cancels GLPR 8739.1B, Software Assurance, dated April 04, 2018.

***LAURENCE SIVIC***

***Digitally signed by LAURENCE SIVIC***
***Date: 2021.08.10 14:47:36 -04'00'***

Laurence A. Sivic
Associate Director

# CHAPTER 1: Software Assurance and Software Safety

*Note: Various personnel in the P/P/F and Safety and Mission Assurance (SMA) organizations at GRC can perform the activities required to satisfy these requirements.*

## 1.1 GRC Project Manager

1.1.1 The responsible GRC Project Manager shall:

a. Comply with requirements in GLPD 7150.2, GRC Software Engineering Requirements, according to the appropriate class of software.

b. Plan and implement software assurance per the processes created by the SMA Quality Engineering and Assurance Branch to plan and implement SASS for their project using the work instructions in the BMS Library, document the lead NASA centers processes followed, or document the processes being followed.

(1) The responsible GRC Project Manager develops and formally documents a tailoring matrix of NASA-STD-8739.8 SASS requirements to capture implementation intent.

(2) Document the SASS processes followed if the project manager chooses not to use GRC SASS processes or if the project is following a different NASA centers processes.

(3) Perform according to the P/P/F software assurance plan and the tailored software assurance and software safety standard requirements in NASA-STD-8739.8.

(4) A NASA-STD-8739.8 NASA SASS Requirements Mapping Matrix template can be requested from grc-swa@mail.nasa.gov.

c. Develop, formally document, and obtain approval of NASA-STD-8739.8 SASS and NPR 7150.2 Requirement Mapping Matrices from the GRC Software Assurance Technical Authority for any tailoring.

(1) The request for relief from a NASA-STD-8739.8 SASS requirement includes the rationale, a risk evaluation, and reference to all material that justifies supporting acceptance in the tailoring matrix. The projects can document their related mitigations and risk acceptance in the approved NASA-STD-8739.8 Requirements Mapping Matrix.

(2) The organization submitting the tailoring request for NASA-STD-8739.8 SASS requirement informs the next higher level of involved management of the tailoring request in a timely manner.

(3) The dispositioning organization reviews the request with the other organizations that could be impacted or have a potential risk (i.e., to safety, quality, cybersecurity, health) with the proposed changes and obtains the concurrence of those organizations.

d. Perform the SASS activities defined in NASA-STD-8739.8 NASA Software Assurance and Software Safety Table 1 per the requirements marked applicable in the NPR 7150.2 requirements mapping matrix for the software component. (NASA-STD-8739.8, SASS-01)

e. Confirm that Form GRC 1707 is completed for all procurements and acquired software or systems containing software subject to NPR 7150.2.

f. Meet the requirements in GLPR 7120.5.30, Space Assurance Requirements for SASS activities for space flight projects.

g. Implement the safety-critical software SASS-01 requirements contained in NASA-STD-8739.8 as appropriate for the software classification and NPR 7150.2 Requirements Mapping Matrix if the software has a safety-critical software determination.

h. Assure there are the necessary resources, including trained personnel, appropriate equipment, and software tools to perform SASS tasks, functions, and requirements.

i. Submit a request to the Quality Engineering and Assurance Branch Chief for SASS support if requesting GRC SASS support from Safety and Mission Assurance (SMA).

j. Perform the following activities if a project decides to not use GRC SMA SASS support:

(1) Inform the Quality Engineering and Assurance Branch Chief of this decision so GRC SMA can maintain SASS's independent reporting channel and ensure required SA training requirements are met by personnel providing SASS support.

(2) Obtain concurrence from the GRC Software Assurance Technical Authority for all software classifications.

(3) Ensure the Software Safety Analysis and Software Safety Critical Determinations are completed by the GRC SMA SA team or independently reviewed by the GRC SMA SA team.

1.1.2 The responsible GRC Project Manager will:

a. Coordinate with software assurance personnel to determine the size and scope of the SASS effort.

(1) Identify who is performing which SASS tasks (contractor, Government, etc.) and who is responsible for audits, deliverables, documentation, and analyses.

(2) Document the agreed-upon activities in project documentation.

b. Assure that a Software Safety Analysis is performed on P/P/F's that plan to have or have a hazard analysis.

(1) The project manager is responsible for the development of the P/P/F software safety analysis and its independent review.

(2) Any differences in software safety's independent software safety critical determinations will be worked through the Engineering Technical Authority and the SMA Technical Authority.

### 1.2 GRC Software Assurance Technical Authority

The GRC Software Assurance Technical Authority (TA) shall:

a. Provide Software Assurance Technical Authority to the P/P/F.

b. Review and provide agreement/disagreement with any tailored NPR 7150.2 and NASA-STD-8739.8 requirements per the completed requirement mapping matrices.

c. Approve NPR 7150.2 and NASA-STD-8739.8 requirement mapping matrices that contain SMA relief or tailoring upon request.

### 1.3 GRC Safety and Mission Assurance (SMA) Technical Authorities

The project SMA Lead, Chief Safety Officer (CSO), or Facility System Safety Program Lead will:

a. Assure that the project(s) complete a thorough hazard analyses which includes software.

b. Review the project's IV&V provider's IV&V Project Execution Plan (IPEP) to ensure it meets NASA IV&V criteria.

c. Ensure that any disagreements between software engineering or the project office and software assurance are identified, reported, tracked, and if not resolved, request a Safety and Mission Assurance Engineering Review Board (SERB) per GLWI-Q-8700.3.

d. Review, ensure, and concur on Software products and processes throughout the project acquisition, development, delivery, operations, and maintenance phases at GRC for the P/P/F.

e. Assure that a Software Safety Analysis is performed on P/P/F's that plan to have or have a hazard analysis.

### 1.4 GRC Project Software Assurance Engineer

The GRC Project Software Assurance Engineer shall:

a. Obtain approval by the appropriate Engineering Technical Authority for any performed software classifications or concur with any engineering software classification of software per the descriptions in NPR 7150.2.

b. Confirm that records of the software Requirements Mapping Matrix (see NPR 7150.2) and each software classification are maintained and updated for the life of the project.

c. Determine if each software component is considered to be safety-critical per the criteria defined in NASA-STD-8739.8 in conjunction with the project.

(1) Document the results in the Software Safety Analysis (SSA) and/or the Software Assurance Plan (SAP), and maintain the SAP/SSA through the software development life-cycle.

(2) Submit the SSA to the appropriate designated SMA Technical Authority for review and concurrence throughout the project acquisition, development, delivery, operations, and maintenance phases.

(3) Any differences in software safety's independent software safety critical determinations will be worked through the designated GRC Engineering Technical Authority and the SMA Technical Authority.

d. Update their SASS or P/P/F plan(s) to fulfill the applicable requirements per the Requirements Mapping Matrix and any approved changes, and initiate adjustments to applicable contracts to meet the modified requirements if a system or subsystem development evolves to meet a higher or lower software classification as defined in NPR 7150.2

e. Report all audit findings and/or high severity software non-conformances to the GRC SMA Software Assurance Discipline Lead at grc-swa@mail.nasa.gov.

f. Plan and implement software assurance, software safety, and IV&V in accordance with GRC SASS processes and work instructions located in the BMS library or document which SASS processes are followed in the Software Assurance or P/P/F plan(s).

g. Collect and maintain project quality and defect metrics for the software, and upon request, communicate data to the GRC SMA Software Assurance Discipline Lead at grc-swa@mail.nasa.gov.

### 1.5 GRC Software Assurance Discipline Lead

1.5.1 The GRC Software Assurance Discipline Lead will:

a. Perform trending and analyses on metrics received by project or software assurance personnel at the GRC center level.

b. Monitor the GRC Software Assurance and Software Safety Workflow mailbox for GRC P/P/F requests.

c. Trends and analyzes quality and defect data as the data is provided by GRC P/P/F(s).

1.5.2 The GRC Software Assurance Discipline Lead shall:

a. Assess opportunities for process improvement on the GRC SASS processes and practices periodically.

b. Document SASS procedures, processes, tools, techniques, and the methods to be used by GRC Safety and Mission Assurance.

### 1.6 GRC requisitioner (i.e., End User)

The GRC requisitioner shall:

a. Complete form GRC 1707 for all procurements and acquired software or systems containing software subject to NPR 7150.2.

b. Contact the project manager regarding required activities defined in this document.

# CHAPTER 2: Software Independent Verification and Validation

## 2.1 GRC Project Manager

The responsible GRC Project Manager shall:

a. Collaborate with the IV&V provider to implement the software IV&V requirements contained in the NASA-STD-8739.8, NASA Software Assurance and Software Safety Standard on projects that are required to have IV&V.

b. Ensure an IV&V Program Execution Plan (IPEP) is developed, negotiated, approved, maintained, and executed if software IV&V is performed on the project.

c. Ensure that IV&V is provided access to development artifacts, products, source code, and data required to perform the IV&V analysis efficiently and effectively if software IV&V is performed on the project.

## 2.2 GRC Project Software Assurance Engineer

The responsible Software Assurance Engineer shall confirm that IV&V requirements contained in NASA-STD-8739.8, NASA Software Assurance and Software Safety Standard are completed or tailored on projects that are required to have IV&V.

## 2.3 GRC Safety and Mission Assurance (SMA) Technical Authorities

The responsible GRC SMA TA updates the Safety and Mission Assurance Plan or Software Assurance Plan to fulfill the applicable requirements per the Requirements Mapping Matrix and any approved changes.

> *Note: The contracting officer or contracting officer representative can be contacted on any potential adjustments or modifications that may be needed to applicable contracts to meet the modified requirements if a system or subsystem development evolves to meet a higher or lower software classification as defined in NPR 7150.2.*

## 2.4 IV&V Provider

The responsible IV&V Provider shall collaborate with the project manager and Safety and Mission Assurance Organization on planning and implementing software IV&V on projects that require it.

# CHAPTER 3: GRC Ground Test Project Software Assurance and Software Safety

*Note: This chapter only applies to projects utilizing GLP-FT-8080.17, Planning and Execution of a Ground Test Project.*

## 3.1 GRC Facility Manager

The responsible GRC Facility Manager shall assist the test customer in developing or including initial software assurance and software safety requirements contained in the NASA-STD-8739.8, NASA Software Assurance and Software Safety Standard on projects as appropriate when evaluating test requests.

## 3.2 GRC Lead Project Engineer

The responsible GRC Lead Project Engineer shall:

a. Assist with overall planning and implementation of NASA-STD-8739.8 NASA Software Assurance and Software Safety requirements.

b. Work with the GRC Ground Test Project Software Assurance Engineer when developing a safety permit package.

## 3.3 GRC Ground Test Project Software Assurance Engineer

3.3.1 The responsible GRC Ground Test Project Software Assurance Engineer shall:

a. Review the safety permit documentation and/or supporting hazard information such as hazard reports, hazard assessments, and hazard analyses to determine if software controls, causes, or mitigates an identified hazard at a minimum.

b. Concur with the projects software classification at a minimum.

c. Independently review and concur with the projects software safety critical determination at a minimum.

d. Independently review that the overall safety plan includes software assurance and software safety if the test or project contains software that has a software safety critical determination at a minimum.

e. Independently review and concur with the Software Safety Analysis if the software is determined by and traceable to hazard analysis at a minimum.

f. Independently review and concur with the Software Assurance Analysis on the detailed software requirements for the ground test project at a minimum.

g. Independently review independent Static Code Analysis results produced by the project at a minimum.

h. Support area safety committees or facility system safety committees for a project or test as requested at a minimum.

3.3.2 The responsible GRC Ground Test Project Software Assurance Engineer supports Test Readiness Review's and panels as requested by the project or test at a minimum.

3.3.3 The responsible GRC Ground Test Project Software Assurance Engineer will communicate metrics, analyses, and trending results to grc-swa@mail.nasa.gov for reference and process improvement at a minimum.

### 3.4 GRC Software Assurance Technical Authority

The GRC Software Assurance TA shall:

a. Provide SASS TA to the P/P/F.

b. Review and provide agreement/disagreement with any tailored NPR 7150.2 and NASA-STD-8739.8 requirement mapping matrices as provided by the P/P/F.

c. Approve NPR 7150.2 and NASA-STD-8739.8 requirement mapping matrices that contain SMA relief or tailoring upon request by the P/P/F.

### 3.5 GRC Hazard Analysis Preparer

The responsible Hazard Analysis preparer will assure that the project(s) complete a thorough hazard analyses which includes software.

### 3.6 Facility System Safety Program Lead

The responsible Facility System Safety Program Lead will assure that the project(s) complete a thorough hazard analyses which includes software.

### 3.7 GRC Program, Project, or Facility Manager

The responsible GRC Project Manager requests SASS support through the Quality Engineering and Assurance (QEA) branch chief if needed.

# Appendix A: Definitions

**Accredit:** The official acceptance of a software development tool, model, or simulation (including associated data) to use for a specific purpose.

**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.

**Analyze:** Review results in-depth, look at relationships of activities, examine methodologies in detail, follow methodologies such as Failure Mode and Effects Analysis, Fault Tree Analysis, trending, and analysis of metrics. Examine processes, plans, products, and task lists for completeness, consistency, accuracy, reasonableness, and compliance with requirements. The analysis may include identifying missing, incomplete, or inaccurate products, relationships, deliverables, activities, required actions, etc.

**Approve:** When the responsible originating official, or designated decision authority, of a document, report, condition, etc. has agreed, via their signature, to the content and indicates the document is ready for release, baselining, distribution, etc. Usually, there will be one "approver" and several stakeholders who would need to "concur" for official acceptance of a document, report, etc. (for example, the project manager would approve the Software Development Plan, but SMA would concur on it.)

**Assess:** Judge results against plans or work product requirements. Assess includes judging for practicality, timeliness, correctness, completeness, compliance, evaluation of rationale, etc., reviewing activities performed, and independently tracking corrective actions to closure.

**Assure:** When software assurance personnel make certain that others have performed the specified software assurance, management, and engineering activities.

**Audit:** (1) systematic, independent and documented process for obtaining audit evidence and evaluating it objectively to determine the extent to which audit criteria are fulfilled (2) independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria (3) independent examination of a software product, software process, or set of software processes to assess compliance with specifications, standards, contractual agreements, or other criteria (4) systematic, independent, documented process for obtaining records, statements of fact, or other relevant information and assessing them objectively, to determine the extent to which specified requirements are fulfilled. *Note:* An audit can be an internal audit (first-party) or an external audit (second party or a third party), and it can be a combined or integrated audit (combining two or more disciplines). Audit results are a clear indication of whether the audit criteria have been met. (IEEE Definition)

**Concur:** A documented agreement that a proposed course of action is acceptable.

**Condition:** (1) measurable qualitative or quantitative attribute that is stipulated for a requirement and that indicates a circumstance or event under which a requirement applies, (2) description of a contingency to be considered in the representation of a problem, or a reference to other procedures to be considered as part of the condition, (3) true or false logical predicate, (4) logical predicate involving one or more behavior model elements, (5) Boolean expression containing no Boolean operators.

**Configuration Item:** An aggregation of hardware, software, or both, that is established and baselined, with any modifications tracked and managed. Examples include requirements document, data block, use case, or unit of code.

**Confirm:** Checks to see that activities specified in the software engineering requirements are adequately done, and evidence of the activities exists as proof. Confirm includes making sure activities are done completely and correctly and have expected content in accordance with approved tailoring.

**Deliverable:** A report or item that has to be completed and delivered under the terms of an agreement or contract. Products may also be deliverables, e.g., software requirements specifications, and detailed design documents.

**Develop:** To produce or create a product or document and to mature or advanced the product or document content.

**Ensure:** When software assurance or software safety personnel perform the specified software assurance and software safety activities themselves.

**Event:** (1) occurrence of a particular set of circumstances (2) external or internal stimulus used for synchronization purposes (3) change detectable by the subject software (4) fact that an action has taken place (5) singular moment in time at which some perceptible phenomenological change (energy, matter, or information) occurs at the port of a unit.

**Hazard:** A state or a set of conditions, internal or external to a system that has the potential to cause harm.

**Hazard Analysis:** Identification and evaluation of existing and potential hazards and the recommended mitigation for the hazard sources found.

**Hazard Control:** Means of reducing the risk of exposure to a hazard.

**Hazardous Operation/Work Activity:** Any operation or other work activity that, without the implementation of proper mitigations, has a high potential to result in loss of life, serious injury to personnel or public, or damage to property due to the material or equipment involved or the nature of the operation/activity itself.

**Independent Verification and Validation (IV&V):** Verification and validation performed by an organization that is technically, managerially, and financially independent of the development organization. (Source: ISO/IEC/IEEE 24765)

**Inhibit:** Design feature that prevents the operation of a function.

**Maintain:** To continue to have; to keep in existence, to stay up-to-date and correct.

**Mission Critical:** Item or function that must retain its operational capability to assure no mission failure (i.e., for mission success). (Source: NPR 8715.3)

**Monitor:** (1) software tool or hardware device that operates concurrently with a system or component and supervises, records, analyzes, or verifies the operation of the system or component; (2) collect project performance data with respect to a plan, produce performance measures, and report and disseminate performance information.

**Perform:** Software assurance does the action specified. Perform may include making comparisons of independent results with similar activities performed by engineering, performing audits, and reporting results to engineering.

**Product:** A result of a physical, analytical, or another process. The item delivered to the customer (e.g., hardware, software, test reports, data), as well as the processes (e.g., system engineering, design, test, logistics) that make the product possible. (Source: NASA-HDBK-8709.22)

**Participate:** To be a part of the activity, audit, review, meeting, or assessment.

**Program:** A strategic investment by a Mission Directorate or Mission Support Office that has a defined architecture and/or technical approach, requirements, funding level, and management structure that initiates and directs one or more projects. A program implements a strategic direction that the Agency has identified as needed to accomplish Agency goals and objectives. (Source: NPR 7120.5)

**Project:** A specific investment having defined goals, objectives, requirements, life cycle cost, a beginning, and an end. A project yields new or revised products or services that directly address NASA's strategic needs. (Source: NPR 7150.2)

**Project Manager:** The entity or individual who accepts the resulting software products. Project managers are responsible and accountable for the safe conduct and successful outcome of their program or project in conformance with governing programmatic requirements. This is usually NASA, but can also refer to the prime contractor-subcontractor relationship as well.

**Software Assurance:** The level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at any time during its life cycle, and that the software functions in an intended manner.

**Provider:** A person or entity that provides something.

**Risk Posture:** A characterization of risk based on conditions (e.g., criticality, complexity, environments, performance, cost, schedule) and a set of identified risks, taken as a whole which allows an understanding of the overall risk, or provides a target risk range or level, which can then be used to support decisions being made.

**Safe State:** A system state in which hazards are inhibited, and all hazardous actuators are in a non-hazardous state. The system can have more than one safe state.

**Safety Critical:** A term describing any condition, event, operation, process, equipment, or system that could cause or lead to severe injury, major damage, or mission failure if performed or built improperly, or allowed to remain uncorrected. (Source NPR 8715.3)

**Safety-Critical Software:** Software is classified as safety-critical if it meets at least one of the following criteria:
a. Causes or contributes to a system hazardous condition/event,
b. Provides control or mitigation for a system hazardous condition/event,
c. Controls safety-critical functions,
d. Mitigates damage if a hazardous condition/event occurs,
e. Detects, reports, and takes corrective action if the system reaches a potentially hazardous state.

**Software:** (1) computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system (2) all or a part of the programs, procedures, rules, and associated documentation of an information processing system (3) program or set of programs used to run a computer (4) all or part of the programs which process or support the processing of digital information (5) part of a product that is the computer program or the set of computer programs. The software definition applies to software developed by NASA, software developed for NASA, software maintained by or for NASA, COTS, GOTS, MOTS, OSS, reused software components, auto-generated code, embedded software, the software executed on processors embedded in programmable logic devices, legacy, heritage, applications, freeware, shareware, trial or demonstration software, and open-source software components. (Source: NPR 7150.2)

**Software Assurance:** The level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at any time during its life-cycle, and that the software functions in an intended manner.

**Software Developer:** A person or thing that develops software, based on program/project requirements.

**Software Life Cycle:** The period that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and sometimes, retirement phase.

**Software Peer Review:** An examination of a software product to detect and identify software anomalies, including errors and deviations from standards and specifications. (Source: IEEE 1028)

**Software Safety:** The aspects of software engineering, system safety, software assurance and software safety that provide a systematic approach to identifying, analyzing, tracking, mitigating, and controlling hazards and hazardous functions of a system where software may contribute either to the hazard(s) or to its detection, mitigation or control, to ensure safe operation of the system.

**Software Validation:** Confirmation that the product, as provided (or as it will be provided), fulfills its intended use. In other words, validation ensures that “you built the right thing.” (Source: IEEE 1012)

**Software Verification:** Confirmation that products properly reflect the requirements specified for them. In other words, verification ensures that “you built it right.” (Source: IEEE 1012)

**Supplier:** A person or organization that provides something needed, such as a software product or service.

**System Safety:** Application of engineering and management principles, criteria, and techniques to optimize safety and reduce risks within the constraints of operational effectiveness, time, and cost.

**Tailoring:** The process used to refine or modify a requirement for a particular project with a justified purpose.

**Track:** To follow and note the course or progress of the product.

# Appendix B: Acronyms

| | |
|---|---|
| APPEL | Academy of Program/Project and Engineering Leadership |
| CAPA | Corrective and Preventative Action |
| CIO | Chief Information Officer |
| CMMI | Capability Maturity Model Integration |
| COTS | Commercial Off The Shelf |
| CSO | Chief Safety Officer |
| DEV | Development |
| DID | Data Item Description |
| EEE | Electrical, Electronic, and Electromechanical |
| GLP | Glenn Procedure |
| GLPD | Glenn Procedural Directive |
| GLPR | Glenn Procedural Requirement |
| GLWI | Glenn Work Instruction |
| GOTS | Government Off The Shelf |
| GRC | Glenn Research Center |
| GSE | Ground Software Equipment |
| HDBK | Handbook |
| IPEP | IV&V Project Execution Plan |
| IV&V | Independent Verification & Validation |
| KDP | Key Decision Point |
| MOTS | Modified Off The Shelf |
| NASA | National Aeronautics and Space Administration |
| NEN | NASA Engineering Network |
| NPR | NASA Procedural Requirement |
| NRRS | NASA Records Retention Schedules |
| OCIO | Office of Chief Information Officer |

| | |
|---|---|
| OSMA | Office of Safety and Mission Assurance |
| OSS | Open Source Software |
| P/P/F | Program/Project/Facility |
| POC | Point Of Contact |
| PPAD | Program and Project Assurance Division |
| PPBE | Planning, Programming, Budgeting, and Execution |
| QE | Quality Engineering |
| QEA | Quality Engineering and Assurance |
| QER | Reliability and System Safety Engineering |
| RFA | Request For Action |
| RID | Review Item Discrepancy |
| SA | Software Assurance |
| SAE | Software Assurance Engineer |
| SAP | Software Assurance Plan |
| SASS | Software Assurance and Software Safety |
| SEPG | Software Engineering Process Group |
| SERB | Safety and Mission Assurance Engineering Review Board |
| SMA | Safety and Mission Assurance |
| SOW | Statement Of Work |
| SSA | Software Safety Analysis |
| STD | Standard |
| STEP | Safety and Mission Assurance Technical Excellence Program |
| TA | Technical Authority |
| WBS | Work Breakdown Structure |

# Change History

| Change | Date | Description/Comments |
|---|---|---|
| Basic | 9/24/2007 | Document converted from CLP (GRC-P2.10.2) to GLPR. |
| A | 2/6/2013 | Document updated to NPR 7150.2A and changes made to address the Quality Audit, Assessment, and Reviews (QAAR) and Requirement Engineering Design Audits and Assessments (REDAA) findings. |
| Change 1 | 7/1/2013 | Added statements 3.1.1 and 4.1.2 to clarify existing requirement in NPR 7150.002 for SMA to perform an independent software safety criticality assessment<br>Added the Applicability Matrix to provide a more detailed mapping of requirements tailoring by class referenced in Appendix F.<br>Appendix changes are as follows:<br>**Appendix G. "**Compliance Matrix" is now the newly added "Applicability Matrix"<br>**Appendix H.** "Software Safety Litmus Test Template" is now the updated "Compliance Matrix"<br>**Appendix I.** "Software Assurance Classification Report Template" is now "Software Safety Litmus Test Template"<br>**Appendix J.** "References" is now "Software Assurance Classification Report Template"<br>**Appendix K**. is now "References"<br>Updated references of appendix throughout the document |
| B | 04/04/2018 | Document updated to reference software assurance processes in SMA, updates made in 7150.2B, and removed Chapters 2 through 4 and Appendices B through K. Removed items were incorporated into previously updated lower-level documents (GLWI). Form updated to the latest GLPR template. Processes and procedures of software assurance are now spelled out in lower level work instructions. |
| C | 08/10/2021 | Major changes due to updates made in NPR 7150.2C. Major changes due to the release of NASA-STD-8739.8A Software Assurance and Software Safety Standard. Major changes due to the cancellation of NASA-STD-8719.13C NASA Software Safety Standard.<br>Updated to meet requirements in GLPR 1410.1 |