SOFTWARE ASSURANCE STANDARD
NASA-STD-2201-93

National Aeronautics and
Space Administration
Washington, DC 20546

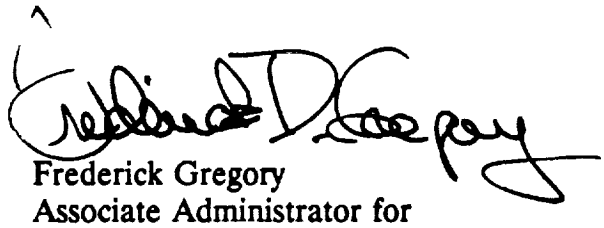
APPROVED: NOVEMBER 10, 1992

PREFACE

This document specifies a standard for the NASA Software Assurance Program that is to be applied by the provider of NASA software. The NASA Software Assurance Standard provides:

- A means for ensuring that required quality is built into NASA software.
- A means for ensuring that software provided to NASA is suitable for its intended purpose.
- A common, tailorable set of activities that comprise a software assurance program.

General questions concerning this publication should be referred to the Office of Safety and Mission Quality, NASA Headquarters, Washington D.C. 20546.



Frederick Gregory
Associate Administrator for
Safety and Mission Quality

INTRODUCTION

The NASA Software Assurance Standard (hereinafter referred to as the "Standard") specifies at a high level an overall NASA Software Assurance Program for software developed for and by NASA. It provides a consistent, uniform basis for specifying and defining the software assurance program to be applied by the developer of NASA software.

This Standard is at the highest level of software assurance standards developed by NASA. Further issuances of NASA software assurance standards will specify in more detail specific processes within the software assurance discipline.

The Standard will have been successfully applied if the:

- NASA software acquirer tailors it to meet the needs and requirements specific to a software development activity.
- Provider of software for NASA develops a software assurance program that meets the tailored requirements.
- Implemented software assurance program reduces the technical and programmatic risks associated with the delivery of software meeting NASA's technical, schedule, and budgetary needs.

The major sections of this standard are as follows:

- Section 1.0, Scope, Purpose, and Application, describes the function and type of activity and provides information on tailoring it to specific projects.
- Section 2.0, References, provides a listing of documents referenced and a glossary of terms used in the Standard.
- Section 3.0, Requirements, contains specific requirements for the software assurance program for providers of NASA software.
- Section 4.0, Quality Assurance Provisions, does not apply to this Standard because it is itself an assurance standard.
- Section 5.0, Packaging, does not apply to this Standard.
- Section 6.0, Additional Information, provides material that is general in nature, but which is not mandatory or contractually binding.

TABLE OF CONTENTS

Section	Title	Page
1.0	Scope, Purpose, and Application	1
1.1	Scope	1
1.2	Purpose	1
1.3	Application	1
1.4	Tailoring	1
2.0	References	3
2.1	Referenced Documents	3
2.2	Glossary	3
2.3	Abbreviations and Acronyms	6
3.0	Requirements	9
3.1	General	9
3.2	Software Assurance	9
3.2.1	Software Assurance Plan	9
3.2.2	Software Assurance Management	10
3.2.3	Software Assurance Records	10
3.2.4	Software Assurance Status Reporting	10
3.2.5	Software Assurance Plan Change Procedures	10
3.2.6	Software Assurance Approval Authority	11
3.3	Software Assurance Functions	11
3.3.1	Software Quality Assurance	11
3.3.2	Software Quality Engineering	12
3.3.3	Software Verification and Validation	12
3.3.4	Nonconformance Reporting and Corrective Action	13
3.3.5	Software Safety Assurance	13
3.3.6	Software Security Assurance	14
3.4	Training	14
3.5	Subcontractor Controls	14
4.0	Quality Assurance Provisions	17
5.0	Packaging	19
6.0	Additional Information	21
6.1	Relationship Between the Acquirer and Provider	21
6.2	Relationship Between Plans, Standards, and Procedures	21

TABLE OF CONTENTS (Continued)

Section	Title	Page
6.3	Acquirer's Assurance	22
6.3.1	Pre-Award	22
6.3.2	Post-RFP, Pre-Award	22
6.3.3	Post-Award, Pre-Development	23
6.3.4	Development	23
6.3.5	Acceptance	24

1.0 SCOPE, PURPOSE, AND APPLICATION

1.1 SCOPE

This Standard specifies the software assurance program for the provider of software. It also delineates the assurance activities for the provider and the assurance data that are to be furnished by the provider to the acquirer. In any software development effort, the provider is the entity or individual that actually designs, develops, and implements the software product, while the acquirer is the entity or individual who specifies the requirements and accepts the resulting products.

1.2 PURPOSE

This Standard specifies at a high level an overall software assurance program for software developed for and by NASA. Assurance includes the disciplines of Quality Assurance, Quality Engineering, Verification and Validation, Nonconformance Reporting and Corrective Action, Safety Assurance, and Security Assurance. The application of these disciplines during a software development life cycle is called Software Assurance. Subsequent lower-level standards will specify the specific processes within these disciplines.

1.3 APPLICATION

The NASA Software Assurance Standard and other NASA standards should be applied to all software developed for and by NASA, including the following:

1.3.1 Deliverable software.

1.3.2 Software included as part of deliverable hardware (including firmware).

1.3.3 Nondeliverable software (commercially available or user-developed) used for development, fabrication, manufacturing process control, testing, or acceptance of deliverable software or hardware (test and acceptance software; software design, test, and analysis tools; compilers, etc.).

1.3.4 Commercially available, reused, or government-furnished software (GFS) designated as a deliverable item or a part thereof.

1.4 TAILORING

This Standard shall be tailored by the acquirer (e.g., NASA program/project manager) in accordance with the classification of the software being developed or acquired. The classification of software is determined by the responsible NASA center or program

office per NMI 2410.10A, "NASA Software Management, Assurance, and Engineering Policy." Tailoring of this Standard consists of the following:

1.4.1 Identifying requirements that are not applicable.

1.4.2 Adding requirements.

1.4.3 Providing quantifiable criteria for the requirements (how often, how many, quality criteria, etc.).

2.0 REFERENCES

2.1 REFERENCED DOCUMENTS

NMI 2410.7A	Assuring the Security and Integrity of NASA Automated Information Resources, July 8, 1988.
NMI 2410.10A	NASA Software Management, Assurance, and Engineering Policy, December 12, 1991.
NHB 1700.1	NASA Basic Safety Manual, Volume 1-B, Chapter 5.
NASA-STD-2100	NASA Software Documentation Standard, July 29, 1991.
SMAP-GB-A201	Software Assurance Guidebook, September 1989, NASA.
SMAP-GB-A301	Software Quality Assurance Audits Guidebook, December 1990, NASA.

2.2 GLOSSARY

Acceptance - Final approval and receipt of a product by its acquirer.

Acceptance Review - Phase transition review for the Acceptance and Delivery life cycle phase.

Acquirer - An organization that obtains a product or capability, such as a software system.

Architectural Design Review - See Preliminary Design Review.

Assurance - Those activities, regardless of the organization conducting the activities, that demonstrate the conformance of a product or process to a specified criteria (such as a functional requirement or a standard).

Assurance Plan - A document containing the technical and planning aspects of the assurance activities for a system/software development or acquisition project.

Audit - An activity to determine through investigation the adequacy of, and adherence to, established standards, procedures, instructions, specifications, or other applicable contractual and licensing requirements, and the effectiveness of implementation.

Baseline - A specification or product that has been reviewed formally and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures.

Configuration Management - Process of identifying and defining the baseline items in a system, controlling the release and change of these items throughout the system/software life cycle, and recording and reporting the status of baseline items and change requests.

Critical Design Review - Phase transition review for the Detailed Design life cycle phase.

Documentation - Any written or pictorial information annotating, describing, defining, specifying, reporting, or certifying activities, requirements, procedures, results, or products.

Firmware - Hardware that contains a computer program and data that cannot be changed in its user environment. The computer program and data contained in the firmware are classified as software; the circuitry containing the computer program and data are classified as hardware.

Formal Inspections - In-process technical reviews of a product of the software life cycle conducted for the purpose of finding and eliminating defects. Formal inspections may be applied to any product or partial product of the software development process, including requirements, design, and code.

Formal Reviews - See phase transition review.

Hardware - Physical equipment used in data processing, as opposed to computer programs, procedures, rules, and associated documentation.

Independent Verification and Validation (IV&V) - A process whereby the products of the software development life cycle phases are independently reviewed, verified, and validated by an organization that represents the acquirer of the software and is completely independent of the provider.

Integration - Process of combining software elements, hardware elements, or both into an overall system.

Life Cycle - Period of time that starts when a software product is conceived and ends when the software is no longer available for use. The traditional "waterfall" software life cycle has eight phases: Concept and Initiation; Requirements; Architectural Design; Detailed Design; Implementation; Integration and Test; Acceptance and Delivery; and Sustaining Engineering and Operations.

Metric - Quantitative measure of extent or degree to which software possesses and exhibits a certain characteristic, quality, property, or attribute.

Nonconformance - A deviation from specified standards, procedures, plans, requirements, or design.

Nonconformance Reporting and Corrective Action - The process used to identify, report, track, and correct nonconformances.

Phase - Period of time during the life cycle of a project in which a related set of software engineering activities is performed. Phases may overlap.

Phase Transition Review - Review at the end of a life cycle phase.

Preliminary Design Review - Phase transition review for the Preliminary (Architectural) Design Life Cycle Phase. Also, known as Architectural Design Review.

Program - NASA Headquarters organization and activities that funds and manages Center-level projects.

Project - Organization and associated activities that produce and/or provide a software system/product.

Provider - Organization that actually develops the software products to the requirements of the acquirer. May be a contractor or an in-house NASA entity.

Quality Assurance - Those assurance activities focused on conformance to standards and procedures.

Quality Engineering - Process of incorporating reliability, maintainability, and other quality factors into software products.

Requirements Analysis - Process of studying user needs to arrive at a specification of system or software requirements.

Requirements Allocation - Process of distributing requirements of a system to subordinate software and hardware elements.

Requirements Review - Phase transition review for the Requirements life cycle phase.

Risk - Combined effect of the likelihood of an unfavorable occurrence and the potential impact of that occurrence.

Risk Management - Process of assessing potential risks and reducing those risks within budget, schedule, and other constraints.

Software - Programs, procedures, rules, and associated documentation and data pertaining to the operation of a computer system. Includes programs and data contained in firmware.

Software Assurance - Planned and systematic set of activities that ensures that software processes and products conform to requirements, standards, and procedures. It includes the disciplines of Quality Assurance, Quality Engineering, Verification and Validation, Nonconformance Reporting and Corrective Action, Safety Assurance, and Security Assurance and their application during a software life cycle.

Test Readiness Review - Phase transition review for the Integration and Test life cycle Phase.

Testing - Process of exercising or evaluating software by manual or automated means to demonstrate that it satisfies specified requirements or to identify differences between expected and actual results.

Verification and Validation - Verification is the process that assures that the software has "been built right"; that is, that each intermediate product during the life cycle meets its specific requirements. Validation is the process of evaluating software to assure that "the right product has been built"; that is, to assure that it meets its functional and performance requirements.

2.3 ABBREVIATIONS AND ACRONYMS

GFS - Government Furnished Software

IV&V - Independent Verification and Validation

NHB - NASA Handbook

NMI - NASA Management Instruction

NRCA - Nonconformance Reporting and Corrective Action

RFP - Request for Proposal

SEB - Source Evaluation Board

SMAP - Software Management and Assurance Program

SOW - Statement of Work

SQA - Software Quality Assurance

SQE - Software Quality Engineering

V&V - Verification and Validation

3.0 REQUIREMENTS

3.1 GENERAL

A software assurance program shall be planned, documented, and implemented for software development activities. The software assurance program shall:

3.1.1 Ensure that assurance requirements are documented and satisfied throughout all phases of the life cycle.

3.1.2 Detect actual or potential conditions that could degrade quality, including deficiencies and system incompatibilities, and provide a process to ensure corrective action is taken and completed.

3.1.3 Assure timely and effective preventive action by identifying root causes of deficiencies and nonconformances.

3.2 SOFTWARE ASSURANCE

3.2.1 SOFTWARE ASSURANCE PLAN

For each development effort, the software assurance process to be applied shall be documented in a software assurance plan in accordance with NASA-DID-M400, contained in the NASA-STD-2100-91, "NASA Software Documentation Standard." The plan shall describe how the activities specified by this assurance standard will be implemented. The software assurance plan shall be reviewed and, if needed, updated at the end of each life cycle phase. The plan shall address, but is not limited to:

3.2.1.1 Descriptions of the procedures for each software assurance task including traceability to assurance program requirements.

3.2.1.2 Description of the role of the software assurance program in activities for continuous improvement such as:

3.2.1.2.1 A strategy that emphasizes prevention, not correction.

3.2.1.2.2 Improved use of tools and techniques.

3.2.1.2.3 Collection and evaluation of metric data.

3.2.1.2.4 Suggestions for improvements in assurance methods.

3.2.2 SOFTWARE ASSURANCE MANAGEMENT

The provider shall designate a software assurance manager who shall be responsible for directing and managing the software assurance program. The software assurance manager shall have a reporting channel to provider management that is independent of the provider's project management and software development function.

3.2.3 SOFTWARE ASSURANCE RECORDS

Records shall be prepared that contain the descriptions and results of all software assurance activities. Results, such as status reports and audit reports, shall include recommended preventative measures and corrective actions. These records shall be available to the acquirer.

3.2.4 SOFTWARE ASSURANCE STATUS REPORTING

Software Assurance status reports shall be prepared by the provider on its software assurance activities on a periodic basis, as specified by the acquirer. The report shall include:

3.2.4.1 Organization and key personnel changes.

3.2.4.2 Assurance accomplishments such as inspection and test activities, reviews, contractor/subcontractor surveys, etc.

3.2.4.3 Subcontractor assurance accomplishments, including items listed above, plus summaries of acceptance and certification reports.

3.2.4.4 Significant problems, their solutions, and remedial and preventive actions.

3.2.4.5 Significant trends in metric data.

3.2.4.6 Recommendations and lessons learned.

3.2.5 SOFTWARE ASSURANCE PLAN CHANGE PROCEDURES

Any proposed deviations from or modifications to the baselined software assurance plan shall be submitted to the acquirer as a formal change request. Proposed changes shall be accompanied by a risk analysis performed to identify the potential impact of the change.

3.2.6 SOFTWARE ASSURANCE APPROVAL AUTHORITY

The software assurance manager shall have approval authority on the establishment and composition of all software baselines and any changes to the baselines.

3.3 SOFTWARE ASSURANCE FUNCTIONS

3.3.1 SOFTWARE QUALITY ASSURANCE

Software Quality Assurance (SQA) is concerned with the evaluation of the quality of, and adherence to, software-related standards and procedures. SQA activities shall be performed during each phase of the software life cycle.

3.3.1.1 The SQA process shall ensure that:

3.3.1.1.1 Standards and procedures for management, engineering, and assurance activities are specified.

3.3.1.1.2 Management, engineering, and assurance adhere to specified standards and procedures.

3.3.1.1.3 All documentation and report formats and content descriptions are defined.

3.3.1.1.4 All plans (configuration management, risk management, assurance plan, management plan, etc.) are completed and implemented according to specified standards and procedures.

3.3.1.1.5 The configuration management process functions according to approved procedures and that all baselined items are maintained under Configuration Management.

3.3.1.1.6 Baseline control, configuration identification, configuration control, configuration status accounting, and configuration authentication activities are carried out.

3.3.1.2 The SQA process shall include the following activities:

3.3.1.2.1 Evaluation of specified standards and procedures.

3.3.1.2.2 Audits of all management, engineering, and assurance processes, for example, Configuration Management.

3.3.1.2.3 Reviews of all project documentation including reports, schedules, and records.

3.3.1.2.4 Monitoring of formal inspections and formal reviews.

3.3.1.2.5 Monitoring/witnessing of formal and acceptance-level software testing.

3.3.2 SOFTWARE QUALITY ENGINEERING

The Software Quality Engineering (SQE) process is concerned with incorporating reliability, maintainability, usability, and similar requirements into the products produced at each phase of the life cycle.

3.3.2.1 The SQE process shall ensure that:

3.3.2.1.1 All quality requirements are defined in a manner that is measurable or otherwise verifiable.

3.3.2.1.2 Quality requirements are considered during design of the software.

3.3.2.1.3 The software is tested/measured to verify compliance with quality requirements.

3.3.2.2 The SQE process shall include the following activities:

3.3.2.2.1 Analysis, identification, and detailed definition of all quality requirements.

3.3.2.2.2 Quality engineering evaluations of standards, design, and code.

3.3.2.2.3 Collection and analysis of metric data pertaining to quality requirements.

3.3.3 SOFTWARE VERIFICATION AND VALIDATION

Software Verification and Validation (V&V) is concerned with ensuring that software being developed or maintained satisfies functional and other requirements and that each phase of the development process yields the right products.

3.3.3.1 V&V activities shall be performed during each phase of the software life cycle and shall include the following:

3.3.3.1.1 Analysis of system and software requirements allocation, verifiability, testability, completeness, and consistency (including analysis of test requirements).

3.3.3.1.2 Design and code analysis including design completeness and correctness.

3.3.3.1.3 Interface analysis (requirements and design levels).

3.3.3.1.4 Formal Inspections.

3.3.3.1.5 Formal Reviews (phase transition reviews).

3.3.3.1.6 Test planning, performance, and reporting.

3.3.3.2 If Independent Verification and Validation (IV&V) is specified, the IV&V activities shall be performed by an organization which reports directly to and is funded directly by the acquirer, and that is other than and independent of, the software provider. The software provider shall cooperate with the IV&V provider and shall facilitate access to all software and assurance products.

3.3.4 NONCONFORMANCE REPORTING AND CORRECTIVE ACTION

Software Nonconformance Reporting and Corrective Action (NRCA) is concerned with reporting, analyzing, and correcting nonconformances and collecting information from which reports on the overall status of nonconformances can be made. NRCA activities shall be performed during each phase of the software life cycle.

3.3.4.1 The NRCA process shall contain, but not be limited to, the following:

3.3.4.1.1 Nonconformance detection and reporting procedures.

3.3.4.1.2 Nonconformance tracking and management procedures.

3.3.4.1.3 Nonconformance impact assessment and corrective action procedures.

3.3.4.1.4 Interfaces to the Configuration Management process.

3.3.5 SOFTWARE SAFETY ASSURANCE

Software Safety Assurance is concerned with the satisfaction of system safety requirements that are allocated to the software, and the identification and verification of adequate safety controls and inhibits that are to be implemented in software.

3.3.5.1 The software safety program shall ensure that:

3.3.5.1.1 Safety-related deficiencies in specifications and design are identified and corrected.

3.3.5.1.2 Software design incorporates positive measures to enhance the safety of the system.

3.3.5.1.3 Software safety is included as an agenda item for formal reviews.

3.3.5.2 The software safety process shall include the following activities:

3.3.5.2.1 Determining the safety criticality for each software component.

3.3.5.2.2 Analyzing consistency, completeness, correctness, and testability of safety requirements.

3.3.5.2.3 Analysis of design and code to ensure that they correctly implement safety-critical requirements.

3.3.5.2.4 Analysis of changes for safety impact.

3.3.5.3 Software safety tasks assigned shall be in compliance with the intent of and the system-related activities required by NHB 1700.1, "NASA Basic Safety Manual." Whenever possible, software safety tasks shall be coordinated with the overall software assurance functions to eliminate duplication of effort.

3.3.6 SOFTWARE SECURITY ASSURANCE

Software Security Assurance is concerned with assuring that security requirements are planned, documented, and implemented during all phases of the software acquisition life cycle. Software security tasks and activities shall include the addressing of security concerns during reviews, analyses, inspections, testing, and audits. Software security tasks shall be in compliance with NMI 2410.7A, "Assuring Security and Integrity of NASA Automated Information Resources."

3.4 TRAINING

Personnel developing and implementing the software assurance process shall be trained and/or experienced in software assurance. Software assurance training shall be obtained and/or originated and maintained as necessary for management, engineering, and assurance personnel. Records shall be maintained and readily available for review of the training, testing, and certification/recertification status of personnel.

3.5 SUBCONTRACTOR CONTROLS

The provider shall be responsible for the adequacy and quality of all software, associated documentation, and services procured through subcontracted efforts. The provider shall

flow down the requirements of this Standard to any subtier provider of software. Provider Software Assurance staff shall ensure acceptable subcontractor performance by:

3.5.1 Developing software assurance requirements for subcontractors.

3.5.2 Participating in the selection of subcontract sources.

3.5.3 Performing/participating in periodic surveys of each subcontractor's facility and software assurance processes to determine their capability of satisfying software requirements.

3.5.4 Reviewing procurement documentation issued at subcontractor facilities, prior to release, for adequacy of software assurance requirements.

3.5.5 Assuring the existence or development of a complete set of software requirements for subcontractor-developed software.

3.5.6 Examining and auditing subcontractor software development and assurance processes and procedures to determine compliance with requirements, standards, and procedures.

3.5.7 Evaluating subcontractor-developed software products for compliance with requirements and standards prior to acceptance.

4.0 QUALITY ASSURANCE PROVISIONS

This section is not applicable to this Standard.

5.0 PACKAGING

This section is not applicable to this Standard.

6.0 ADDITIONAL INFORMATION

This section contains additional information pertaining to this Standard. The material provided is intended for informative purposes only and should not be considered additional requirements. Further information on Software Assurance can be found in NASA Guidebooks such as SMAP-GB-A201, "Software Assurance," and SMAP-GB-A301, "Software Assurance Audits."

The following paragraphs explain the acquirer's role in software assurance. Paragraphs 6.1 and 6.2 provide background information on some key concepts. Paragraph 6.3 describes the software acquisition process and the role of the acquirer's assurance efforts in that process.

6.1 RELATIONSHIP BETWEEN ACQUIRER AND PROVIDER

This Standard defines two types of organizations, acquirers and providers. The acquirer is the organization that is purchasing the software, product, or associated services. Usually, the acquirer is NASA or an organization within the Agency. The provider is the organization that is delivering that product or service. This can be a contractor, a separate NASA organization, or, in some cases, the acquirer and the provider are the same organization. Providers may provide software or services such as IV&V, consulting, etc. A provider is ultimately responsible for providing a product or service to the acquirer according to the acquirer's requirements. The acquirer is ultimately responsible for the quality and effectiveness of all products.

The material in Section 3 of this Standard can be used by both an acquirer and a provider. Both organizations will perform the same types of activities and manage their assurance programs in a similar manner.

6.2 RELATIONSHIP BETWEEN PLANS, STANDARDS, AND PROCEDURES

Software plans (i.e., management plans, assurance plans, risk management plans) are prepared in conformance with specified documentation standards. The purpose of these plans is to document/specify the conduct of all activities. The plans will specify standards and procedures to be used by both the acquirer and the provider. The plans, once reviewed and approved by the designated authority, are used to determine what to assure and how to perform the assurance.

Standards and procedures establish the methods by which software will be managed, engineered, and assured, and are the criteria against which the products and processes will be measured. Standards and procedures, once selected and approved, must be tailored to the needs of each project before they are imposed. Tailoring of standards and procedures is the responsibility of the organization imposing them.

6.3 ACQUIRER'S ASSURANCE

The majority of software systems used at NASA are developed with NASA as the acquirer and a contractor as a provider. A typical scenario of how the acquirer should assure the project is given in the following paragraphs. Modifications to this scenario are needed if NASA also is a provider.

6.3.1 PRE-AWARD

Before awarding a contract for the project, the acquirer will create its requirements, management plan, assurance plan, and Request for Proposal (RFP) which includes a Statement of Work (SOW). These documents will be produced according to specified standards and/or procedures.

The acquirer should ensure that these standards and procedures are specified, tailored, and implemented by the acquirer and that the acquirer's requirements, management plan, and RFP are complete and adequate. The acquirers should particularly check the assurance requirements in the RFP. Specifically, all assurance requirements should be determined and related standards and procedures to be imposed on the provider should be selected, tailored, and included in the RFP.

The primary pre-award assurance activities carried out are reviews of the acquirer's requirements, management plan, and RFP. Assurance personnel should participate in all review and Source Evaluation Board (SEB) activities. The assurance manager should sign off on the acquirer's requirements, management plan, and the RFP.

Additionally, the acquirer's preliminary assurance plan should be completed. This plan should cover, as a minimum, all activities that will be performed before the awarding of the contract and all other activities not dependent on specific provider responses to the RFP.

6.3.2 POST-RFP, PRE-AWARD

Once proposals have been received from potential contractors, the acquirer evaluates the proposals in preparation for contract award. Pre-award surveys of prospective providers may be performed to fully assess their capabilities. The contract is then awarded.

Assurance personnel should participate in proposal evaluations, pre-award surveys, and review and approval of the contract. Assurance personnel should pay particular attention to prospective providers' assurance capabilities during evaluations and surveys.

6.3.3 POST-AWARD, PRE-DEVELOPMENT

Once the contract is awarded, the acquirer's assurance plan should be completed. Many details of an assurance plan cannot be completed until the provider is determined and the formal contract fixed. This is due to many reasons, including:

6.3.3.1 Standards and procedures to be used by the provider may not be finalized until the contract is final.

6.3.3.2 Assurance requirements may not be finalized until the contract is awarded.

The provider will prepare their preliminary management and assurance plans for acquirer review and approval.

6.3.4 DEVELOPMENT

During development, the software requirements are defined, the designs are finalized, the software is coded, and the software system is integrated and tested. The acquirer should be primarily concerned with:

6.3.4.1 Reviewing the provider's assurance process status reports regularly to ensure that the provider is performing all specified activities.

6.3.4.2 Participating in all end-of-phase reviews.

6.3.4.3 Participating on the Change Control Board.

6.3.4.4 Reviewing proposed changes to the acquirer's assurance plan.

6.3.4.5 Reviewing the acceptance test plan.

Additionally, the acquirer's assurance process should contain some activities to ensure that the provider is adhering to approved plans and procedures and that these plans and procedures are effectively fulfilling their purpose. These activities may include audits, reviews, analyses, etc.

The acquirer's assurance process must also include oversight and evaluation of the acquirer's management and engineering processes. Specifically, reviews, audits, and evaluations should be performed to ensure adherence to and effectiveness of approved plans and procedures.

6.3.5 ACCEPTANCE

During acceptance, the acquirer verifies that the software and all related products (code, documentation, etc.) are complete and that they meet all of the specified requirements.

The acquirer will:

6.3.5.1 Review a functional demonstration of the software conducted by the provider and/or perform acceptance testing of the software to assure that it meets its requirements.

6.3.5.2 Review all acceptance and delivery documentation for completeness and accuracy.

The acquirer must ensure that any acquirer facilities (buildings, hardware, etc.) are prepared to receive and implement use of the software.