ISRO SOFTWARE PROCESS DOCUMENT (ISPD)

OCTOBER 2006
INDIAN SPACE RESEARCH ORGANISATION

FOREWORD

Software is an integral part of all major systems and plays a vital role in all the critical activities of ISRO. The need for a common software engineering standard in ISRO/DOS was felt in the early nineties, and ISRO Software Engineering Standard (ISES-92) was established. As the application areas and complexity of software activities increased, the adequacy of the above standard to cope up with the new requirements was recently reviewed. It is now decided that IEEE-12207 standard be followed for all software activities in ISRO/DOS.

The implementation of the software standard in ISRO/DOS is supervised by ISRO Software Control Board (ISCB). Necessary guidelines and procedures are generated under the guidance of this Board and the progress of implementation will be continuously monitored for improvement and updating. The representatives from the centres and ISCB have done a commendable job in reviewing the current practices and arriving at a standard set of procedures to be followed.

ISCB has now finalized the "ISRO Software Process Document" (ISPD), which will serve as the apex document in line with IEEE-12207 standard for all the centres and units of ISRO/DOS. This document includes definition of software life cycles for the different classes of software within ISRO/DOS and also the linkage of different activities and tasks to IEEE-12207 standard. ISPD also identifies the activities and tasks considered as mandatory at this stage of implementation of the standard. It is intended that all centers and units of ISRO/DOS streamline their software activities in line with this document.

(G. Madhavan Nair)
Chairman, ISRO / Secretary, DOS

ISRO Software Control Board (ISCB)

1. Dr R.Vikraman Nair, VSSC	…	Chairman
2. Shri P.J.Bhat, ISAC	…	Vice-Chairman
3. Shri V.Seshagiri Rao, SDSC	…	Member
4. Dr R.Ramakrishnan, SAC	…	Member
5. Smt B.Lakshmi,NRSA	…	Member
6. Dr C.V.S.Praksah, ANTRIX	…	Member
7. Shri Y.Krishna Chandra, ISTRAC	…	Member
8. Shri H.D.Ananda,MCF	…	Member
9. Smt. B.Valsa, VSSC	…	Member
10. Shri J.M.Bhaskaran,LPSC	…	Member
11. Shri S.Hemachandran,IISU	…	Member
12. Shri B.V.Prasad,ISAC	…	Member
13. Dr. Dominic Damien, ISRO HQ	…	Member
14. Shri K.Parthasarathy, ISRO HQ	…	Member-Secretary

ISPD Sub-Committee

1.	Smt B.Lakshmi	NRSA
2.	Sri B.V.Prasad	ISAC
3.	Sri Krishna Chandra	ISTRAC
4.	Sri Y R Srinivas Anand	SDSC
5.	Smt V.Jayanthy	VSSC
6.	Smt B.Valsa	VSSC

Software Standards Editorial Committee (SSEC)

1.	Shri N.Jayalal	VSSC
2.	Smt K.S. Prasadakamari	ISAC
3.	Shri P.Sunil	SDSC
4.	Shri Rajesh Khandelwal	SAC
5.	Shri K.Parthasarathy	ISRO HQ

Other Contributors

1.	Shri N.Jayalal	VSSC
2.	Shri W.V.Eswar Praksah	ISAC
3.	Shri Subramanya Udupa	ISAC
4.	Smt U.N.Vasantha Kumari	ISAC
5.	Shri M.G.Raykar	ISAC
6.	Shri P J C Reddy	SDSC
7.	Shri P.Sunil	SDSC
8.	Shri D.Muralikrishna	SDSC
9.	Shri R.L.N.Murthy	ANTRIX

TABLE OF CONTENTS

THIS PAGE IS INTENTIONALLY LEFT BLANK

1

Introduction

1.1 Purpose

ISRO/DOS Projects are planned towards the design and development of space missions related to applications of national importance. The design and development of systems involved in these missions play an important role in achieving the objectives of the mission. Software is an integral part of such systems. Since the application areas and complexity of software have increased considerably, a need was felt in developing a new software process framework for ISRO/DOS in line with international standards.

ISRO Software Control Board (ISCB), the apex body for ISRO software standardization, has generated this document, ISRO Software Process Document(ISPD) which is organized in line with IEEE/EIA:12207. This document which is an implementation guide of IEEE/EIA:12207 software framework , prescribes a set of processes and related plans, procedures, guidelines and templates taking into consideration existing software practices in centres of ISRO/DOS. It is applicable to all software activities in ISRO/DOS.

ISPD describes in detail the architectural framework for software life cycle activities. It also identifies the entry and exit criteria for the various software lifecycle processes and essential documents to be generated as the outcome of different process activities. The users of this document are responsible for selecting a lifecycle model for a specific software project based on the development strategy, and mapping the processes, activities and tasks onto the model to be adopted. The users are also responsible for selecting and applying appropriate software development methods suitable for the project.

1.2 Scope

ISPD defines the software process framework to be followed for different categories of software in ISRO/DOS.

Space systems in ISRO/DOS include spacecrafts, launch vehicles, scientific payloads and the associated ground equipments and facilities. To address software systems related to the above areas, software in ISRO/DOS is classified into following broad categories

- **Onboard** - **Used in on-board applications**
- **Checkout and Simulation** - **Used in checkout systems and simulation test beds**
- **Launch Support & Test Facilities** - **Used for launch activities, test facilities and process control**
- **Mission** - **Used for satellite and vehicle mission activities**
- **Image Processing** - **Used for processing remote sensing data**
- **Information Services** - **Used for information systems and services**
- **Scientific** - **Used for design, analysis and other studies**

This document is applicable to the above-mentioned categories of software.

Throughout this document an effort is made to use ‘shall’ to express a provision that is binding between two or more parties, ‘will’ to express a declaration of purpose or intent by one party, ‘should’ to express recommendations among other possibilities and ‘may’ to indicate a course of action permissible within the limits of ISPD.

2

References

2.1 IEEE/EIA 12207.0-1996: (ISO/IEC 12207) Standard for Information Technology—Software Life Cycle Processes

2.2 IEEE/EIA 12207.1-1996: (ISO/IEC 12207) Standard for Information Technology—Software Life Cycle Processes-Life cycle data

2.3 IEEE/EIA 12207.2-1997: (ISO/IEC 12207) Standard for Information Technology—Software Life cycle Processes-implementation consideration

3

Definitions / Abbreviations

3.1 Definitions

Acquirer - An Organisation/Entity/Area/Group/Division that procures a system, software product or software service from a supplier.

Audit - An independent assessment of software products and processes to assess compliance with requirements.

Baseline - A formally approved designated version of a configuration item.

Process - A set of interrelated activities, which transform inputs into outputs.

Software Project - The set of work activities, both technical and managerial, required satisfying the terms and conditions of a project agreement. A software project should have specific starting and ending dates, well-defined objectives and constraints, established responsibilities, and a budget and schedule.

Software Configuration Item - An entity within a configuration that satisfies an end use function and that can be uniquely identified at a given reference point.

Software unit - Smallest compilable piece of code, which cannot be decomposed further into compilable segments.

Software Component - Comprises of software units that can be coded, compiled and tested.

Supplier - An Organisation/Entity/Area/Group/Division that enters into a contract with the acquirer for the supply of a system, software product or software service under the terms of the contract.

System - An integrated composite that consists of one or more of hardware, software, facilities and people that provides a capability to satisfy a stated need or objective.

User Manual - Document that describes the interaction of the people with the system for operation.

Validation - A process to ensure that the requirements baseline functions and performances are correctly and completely implemented in the final product.

Verification - A process to establish that adequate specifications and inputs exists for any activity, and the outputs of the activities are correct and consistent with the specifications and input.

3.2 Abbreviations

CM	-	Configuration Management
CMB	-	Configuration Management Board
FRD	-	Formulation Requirements Document
ICD	-	Interface Control Document
IRD	-	Interface Requirements Document
ISPD	-	ISRO Software Process Document
MIS	-	Management Information System
QMS	-	Quality Management System
RFI	-	Request for information
RFP	-	Request for Proposal
SAD	-	Software Architectural Design Document
SCMP	-	Software Configuration Management Plan
SDD	-	Software Design Document
SRS	-	Software Requirements Specification
SQA	-	Software Quality Assurance
SQAP	-	Software Quality Assurance Plan
SRD	-	System Requirements Document
SDP	-	Software Development Plan
SMP	-	Software Project Management Plan
SMNP	-	Software Maintenance Plan
STP	-	Software Test Plan
SPQA	-	Software Product Quality Audit
SVVP	-	Software Verification and Validation Plan
VP	-	Validation Plan
V&V	-	Verification and Validation

4

Organisation of ISPD

4.1 Document Organisation

ISPD is structured into following two major parts

- **Software process framework description**
- **Applicability of software process framework to different categories of software developed in ISRO/DOS**

The processes that are to be followed as part of the life cycle activities are described in detail from Sec 5 to Sec 7. Sec 8 provides the following details for different categories of software

- **Sequence Charts for process flow**
- **Activity Linkage table which provides the mapping of ISPD to IEEE/EIA 12207 tasks with pointers to agency responsible for performing them.**

Two tables (Table-1, Table-2) provide the applicability of IEEE/EIA 12207 processes to different categories of software identified

Table-1 provides the applicability matrix for activities under Acquisition and Supply processes of ISPD to software, which may be outsourced, developed within a centre (Intracentre) and developed jointly by teams in different centres (Intercentre). This is applicable to all categories of software identified. Table-2 provides the applicability of tasks to Development, Operation, Maintenance and Support activities. The following three attributes are used in Table-1 and Table-2.

- **Mandatory (M) : Shall be followed for the particular task**
- **Tailorable(T) : May be followed for the task with required tailoring**
- **Not Applicable (NA) : Not relevant for the particular task**

The tailoring criteria / rationale shall be included in the Software Project Management Plan (SMP) and the record shall be maintained as part of SMP.

Annexure – A gives the list of available IEEE standards for the documentation. For generating documents according to these templates, the corresponding IEEE Guides are to be referred.

4.2 Software Life Cycle Processes

Software life cycle processes in ISPD are classified as follows

- **Primary life cycle processes**

  Processes that are applicable to the acquisition, supply, development, operation and maintenance of the software.

- **Supporting life cycle processes**

  Processes that support another process as an integral part with a distinct purpose and contribute to the successes and quality of the software. A supporting process is employed and executed, as needed, by another process. These include processes related to Documentation, Verification, Validation, Configuration Management, Software Quality Assurance, Audit, Joint Reviews and Problem Resolution.

- **Organizational life cycle processes**

  Processes that are employed by the organization to effectively establish and implement the life cycle processes, resources and their continuous improvements. These include processes related to Management, Infrastructure, Training and Improvement.

To ensure effective planning and implementation, each life cycle process is broken into a set of activities and tasks, each having its associated milestones. A diagrammatic representation covering the software lifecycle processes described in this document is shown in Fig 1.

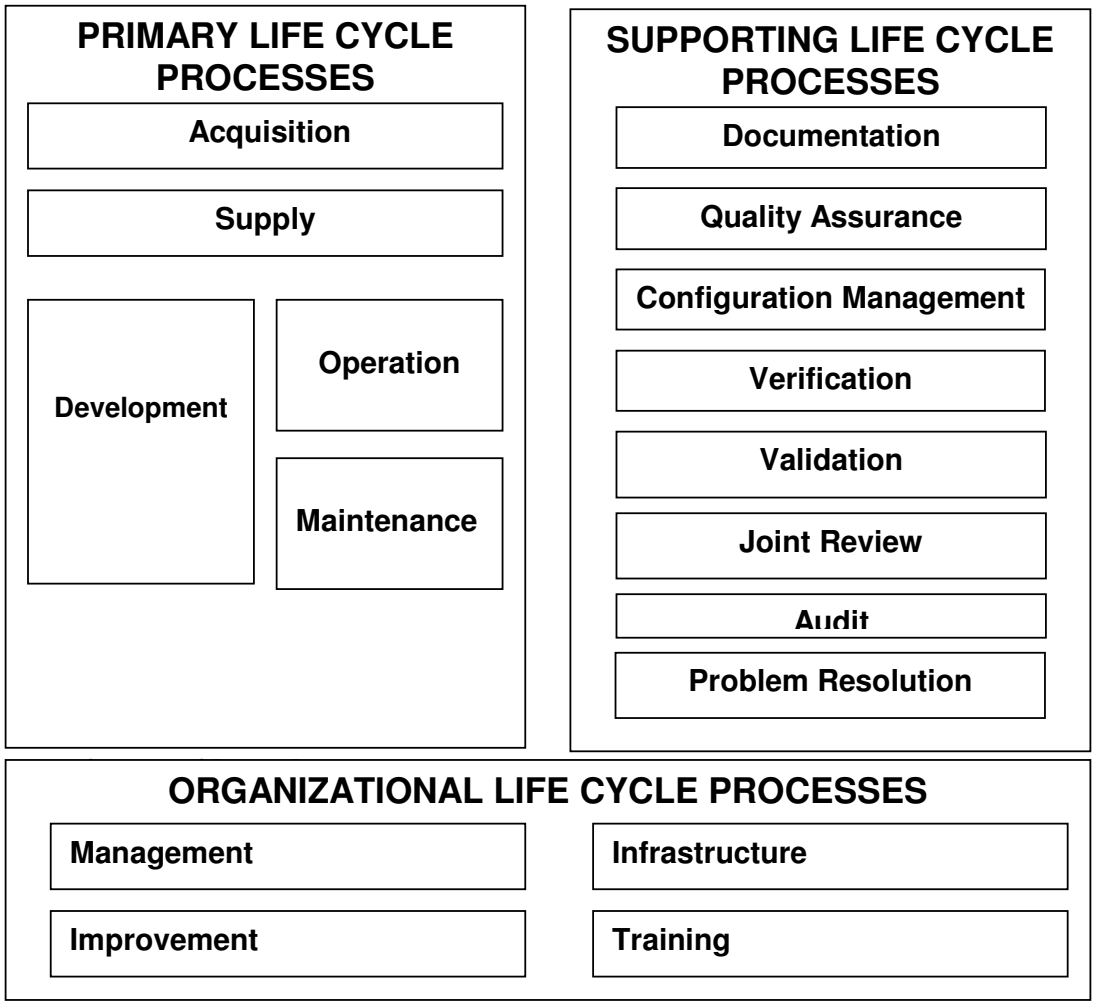


Fig 1: Software Life Cycle Processes

5

Primary Life Cycle Processes

The primary life cycle processes consist of five processes, namely

- **Acquisition Process**
- **Supply Process**
- **Development Process**
- **Operation Process**
- **Maintenance Process.**

5.1 Acquisition Process

Entry Criteria: ***Project Start Up***

This process defines the activities and tasks of the acquirer, which are to be performed during the acquisition of a software product or software service. The acquirer could be Organisation, Entity or Division. This process consists of the following activities:

- **Generation of indent proposal**
- **Request-For-Proposal preparation**
- **Contract preparation and update**
- **Supplier monitoring**
- **Acceptance and completion**

Exit Criteria : ***Receipt of Product /Service meeting the acceptance criteria***

5.1.1 Generation of indent proposal / Request for information-RFI

Required Input: ***Acquisition Need***

5.1.1.1 The Acquirer begins the acquisition process by describing a concept or a need to acquire, develop, or enhance a system, software product or software service and generate an RFI.

5.1.1.2 The acquirer will define and analyze the system requirements, which should include business, organizational, user as well as safety, security, and other criticality requirements along with related design, testing, and compliance standards and procedures. The Acquirer may release a RFI detailing the draft acquisition plan to possible suppliers. The Acquirer shall establish a vendor selection process using responses to RFI.

5.1.1.3 If the Acquirer retains a supplier to perform system requirements analysis, the acquirer will approve the analyzed requirements.

5.1.1.4 The acquirer may perform the definition and analysis of software requirement or retain a supplier to perform this task.

5.1.1.5 The Development Process (5.3) should be used to perform the tasks in 5.1.1.2 and 5.1.1.4.

5.1.1.6 The Acquirer will consider options for acquisition against analysis of appropriate criteria to include risk, cost and benefits for each option, which include:

a) Purchase an off-the-shelf software product that satisfies the requirements.
b) Develop the software product or obtain the software service internally.
c) Develop the software product or obtain the software service through contract.

d) A combination of a, b, and c above.
e) Enhance an existing software product or service.

5.1.1.7 When an off-the-shelf software product is to be acquired, the following conditions are to be satisfied:

a) Requirements for the software product
b) Documentation availability
c) Proprietary, usage, ownership, warranty and licensing rights
d) Planning of future support for the software product.

5.1.1.8 The Acquirer should prepare, document and execute an acquisition plan, which should contain the following:

a) Requirements for the system;
b) Planned employment of the system;
c) Type of contract to be employed;
d) Responsibilities of the organizations involved;
e) Support concept to be used;
f) Risks considered as well as methods to manage the risks.

5.1.1.9 The acceptance strategy and conditions (criteria) are to be defined and documented.

Expected Output: ***Acquisition Plan***

5.1.2 Request-for-proposal/Tender preparation.

Required Input: ***Acquisition Plan***

5.1.2.1 The acquirer should document the acquisition requirements, the content of which depends upon the acquisition option selected in 5.1.1.6. The acquisition documentation should include, as appropriate:

a) System requirements
b) Scope statement
c) Instructions for bidders
d) List of software products
e) Terms and conditions
f) Control of subcontracts
g) Technical constraints

5.1.2.2 The acquirer should determine which processes, activities, and tasks of the standard (Table-1) are appropriate for the project. The acquirer should specify the applicable supporting processes (section 6) and their performing organizations, including responsibilities (if other than supplier), so that the suppliers may, in their proposals, define the approach to each of the specified supporting processes. The acquirer will define the scope of those tasks that reference the contract.

5.1.2.3 The acquisition documentation (RFP) will also define the contract milestones at which the supplier's progress will be reviewed and audited as part of monitoring the acquisition (6.6 and 6.7)

5.1.2.4 The acquisition requirements should be given to the organization selected for performing the acquisition activities.

Expected Output: ***Request-For-Proposal (Tender)***
Responses to RFP

5.1.3 Contract preparation and update

Required Input: ***Request-For-Proposal (Tender)***
Responses to RFP

5.1.3.1 The acquirer should establish a procedure for supplier selection including proposal evaluation criteria and requirements compliance weighting.

5.1.3.2 The acquirer should select a supplier based upon the evaluation of the suppliers' proposals, capabilities, and other factors that need to be considered.

5.1.3.3 The Acquirer will involve other parties, including potential suppliers, before contract award, in tailoring the clauses of ISPD for the project. However, the acquirer will decide the final tailoring strategy.

5.1.3.4 The acquirer will then prepare and negotiate a contract with the supplier that addresses the acquisition requirements, including the cost and schedule, of the software product or service to be delivered. The contract will address proprietary, usage, ownership, warranty and licensing rights associated with the reusable off-the-shelf software products.

5.1.3.5 Once the contract is underway, the acquirer will control changes to the contract through negotiation with the supplier as part of a change control mechanism. Changes to the contract shall be investigated for impact on project plans, costs, benefits, quality, and schedule.

Expected Output: ***Purchase Order/ Supplier Contract***

5.1.4 Supplier monitoring

Required Input: ***Purchase order/ Supplier Contract***

5.1.4.1 The acquirer shall monitor the supplier's activities in accordance with Joint Review Process(6.6) and the Audit Process (6.7). The acquirer should supplement the monitoring with the Verification Process (6.4) and the Validation Process (6.5) as needed.

5.1.4.2 The acquirer will co-operate with the supplier to provide all necessary information in timely manner and resolve all pending items.

Expected Output: ***Evaluation Report***

5.1.5 Acceptance and completion

Required Input: ***Evaluation report and contract***

5.1.5.1 The acquirer should prepare for acceptance based on the defined acceptance strategy and criteria. The preparation of test cases, test data, test procedures, and test environment should be included. The extent of supplier involvement should be defined.

5.1.5.2 The acquirer will conduct acceptance review and acceptance testing of the deliverable software product or service and will accept it from the supplier when all acceptance conditions are satisfied. The acceptance procedure should comply with the provisions of 5.1.1.9.

5.1.5.3 After acceptance, the acquirer should take the responsibility for the configuration management of the delivered software product (6.2).

Expected Output: ***Acceptance Report***
Accepted Product/Service

5.2 Supply Process

Entry Criteria: ***Proposal Initiation***
RFI from acquirer

The Supply Process contains the activities and tasks of the supplier. The process may be initiated either by a decision to prepare a proposal to answer an acquirer's request for proposal or by signing and entering into a contract with the acquirer to provide the system, software product or software service.

The following activities are addressed in this process:

- **Initiation**
- **Preparation of response**
- **Contract**
- **Planning**
- **Execution and control**
- **Review and evaluation**
- **Delivery and completion**

Exit Criteria: ***Delivery of Product /Service***

5.2.1 Initiation

5.2.1.1 The supplier conducts a review of requirements in the request for proposal taking into account organizational policies and other regulations.

5.2.1.2 The supplier should make a decision to bid or accept the contract.

5.2.2 Preparation of response.

5.2.2.1 The supplier should define and prepare a proposal in response to the request for proposal, including its recommended tailoring.

5.2.3 Contract

5.2.3.1 The supplier shall negotiate and enter into a contract with the acquirer organization to provide the software product or service.

5.2.3.2 The supplier may request modification to the contract as part of the change control mechanism.

5.2.4 Planning

Required Input: ***Project Initiation / Contract***

5.2.4.1 The supplier shall conduct a review of the acquisition requirements to define the framework for managing and assuring the project and for assuring the quality of the deliverable software product or service.

5.2.4.2 If not stipulated in the contract, the supplier shall define or select a software life cycle model appropriate to the scope, magnitude, and complexity of the project. The processes, activities, and tasks of software category shall be selected and mapped onto the life cycle model from Table-2.

5.2.4.3 The supplier shall establish requirements for the plans for managing and assuring the project and for assuring the quality of the deliverable software product or service. Requirements for the plans should include resource needs and acquirer involvement.

5.2.4.4 Once the planning requirements are established, the supplier shall consider the options for developing the software product or providing the software service, against an analysis of risks associated with each option. Options include:

a) Develop the software product or provide the software service using internal resources.

b) Develop the software product or provide the software service by subcontracting.
c) Obtain off-the-shelf software products from internal or external sources.
d) A combination of a, b, and c above.

5.2.4.5 The supplier shall develop and document project management plan(s) based upon the planning requirements and options selected in 5.2.4.4. Items to be considered in the plan include but are not limited to the following:

a) Project organizational structure and authority and responsibility of each organizational unit, including external organizations
b) Engineering environment, including test environment, library, equipment, facilities, standards, procedures, and tools
c) Work breakdown structure of the life cycle processes and activities, including the software products, software services and non-deliverable items, to be performed
d) Management of the quality characteristics
e) Management of the safety, security, and other critical requirements f) Subcontractor management, including subcontractor selection and involvement between the subcontractor and the acquirer, if any;
f) Quality assurance (6.3)
g) Verification (6.4) and validation (6.5) including the approach for interfacing with the verification and validation agent, if specified
h) Acquirer involvement by such means as joint reviews (6.6), audits (6.7), informal meetings, reporting, modification and change; implementation, approval, acceptance, and access to facilities
i) User involvement
j) Risk management
k) Security policy
l) Approval required by such means as regulations, required certifications, proprietary, usage, ownership, warranty and licensing rights
m) Means for scheduling, tracking, and reporting
n) Training of personnel (7.4)

Note: Separate plans for quality, safety and security may be developed.

Expected Output: ***Software Project Management Plan (SMP)***

5.2.5 Execution and Control

Required Input: ***Software Project Management Plan (SMP)***

5.2.5.1 The supplier shall implement and execute the project management plan(s) and the plans may be updated/modified as needed to reflect changes in requirements, policies and procedures.

5.2.5.2 The supplier shall:

a) Develop the software product in accordance with Development Process (5.3).
b) Operate the software product in accordance with Operation Process (5.4).
c) Maintain the software product in accordance with Maintenance Process (5.5).

5.2.5.3 The supplier shall monitor and control the progress and quality of the software products or services of the project throughout the contracted life cycle. This shall be an ongoing, iterative task, which shall provide for:

a) Monitoring progress of technical performance, costs, and schedules and reporting of project status
b) Problem identification, recording, analysis, and resolution

5.2.5.4 The supplier shall manage and control the subcontractors in accordance with the Acquisition Process (5.1). The supplier shall pass down all contractual requirements necessary to ensure that the software product or service delivered to the acquirer is developed or performed in accordance with the prime-contract requirements.

5.2.5.5 The supplier shall interface with the independent verification, validation, or test agent as specified in the contract and project plans.

5.2.5.6 The supplier shall interface with other parties as specified in the contract and project plans.

Expected Output: ***Navigation chart overlaid with Monitoring Result***

5.2.6 Review and Evaluation

Required Input: ***Contract /Purchase order***

5.2.6.1 The supplier should co-ordinate contract review activities

5.2.6.2 The supplier should conduct or support the informal meetings, acceptance review, acceptance testing, joint reviews and audits as specified in the project plans. The joint reviews shall be conducted in accordance with 6.6, audits in accordance with 6.7.

5.2.6.3 The supplier shall perform verification and validation to demonstrate the compliance of the software product/service with the requirements quality assurance activities

5.2.6.4 The supplier shall make available to the acquirer the reports of evaluation, reviews, audits, testing, and problem resolutions as specified in the contract

5.2.6.5 The supplier shall provide the acquirer access to the supplier's and subcontractors' facilities for review of software products or services as specified in the contract and project plans.

5.2.6.6 The supplier shall perform quality assurance activities in accordance with 6.3.

Expected Output: ***Review Records / Test Results***

5.2.7 Delivery and Completion

Required Input: ***Review Records / Test Results***

5.2.7.1 The supplier shall deliver the software product or service as specified in the contract.

5.2.7.2 The supplier shall provide assistance to the acquirer in support of the delivered software product or service as specified in the contract.

Expected Output: ***Delivered Product / Services***

5.3 Development Process

Entry Criteria: ***Software Project Management Plan (SMP)***

The Development Process contains the activities and tasks that are to be performed during the development of software. The activities of this process shall be selected and mapped onto the life cycle model selected. These activities may overlap or interact and may be performed iteratively. The activities may be carried out as per guidelines approved for the respective category of software.

The following activities are addressed in this process:

- **Development Planning**
- **System Requirements Analysis**
- **System Architectural Design**
- **Software Requirements Analysis**
- **Software Architectural Design**
- **Software Detailed Design**
- **Software Coding and Testing**
- **Software Integration**
- **Software Qualification Testing**
- **System Integration**
- **System Qualification Testing**
- **Software Installation**
- **Software Acceptance support**

For a system including Hardware & Software (or, embedded systems), all the above activities shall be addressed. Here the software is treated as an integral part of the total system and performs certain functions in that system. This is implemented by extracting the software requirements from the system requirements and developing the software and integrating it into the system.

The role of developer varies depending on the type of system developed. If the software is a part of hardware-software system, the developer takes part in system activities as described in the plans. If the software possibly with its computer is considered to constitute a system, or only the software in an

existing software-hardware is changed, the developer is responsible for performing or supporting these activities.

In the system related activities namely 5.3.2, 5.3.3, 5.3.10 and 5.3.11 the developer is tasked to perform or support these as specified in the contract. As mentioned in the acquisition process(5.1), the acquirer may elect to perform the activity 5.3.2. Similarly the acquirer may elect to perform activities 5.3.3, 5.3.10 and 5.3.11 or task the developer to perform these activities or employ other organizations to perform them.

Exit Criteria: ***Product/Service Acceptance***

5.3.1 Development Planning

Required Input: ***Software Project Management Plan (SMP)***

5.3.1.1 An appropriate life cycle model shall be established before continuing with the development process. The activities and tasks of the Development process shall be mapped onto the life cycle model.

5.3.1.2 The developer shall:

a) Document the outputs in accordance with the Documentation Process(6.1).
b) Place the outputs under the Configuration Management Process (6.2) and perform change control in accordance with it.
c) Document and resolve problems and nonconformance found in the software products and tasks in accordance with the Problem Resolution Process (6.8).
d) Perform the supporting processes (section 6) as specified in the contract.

5.3.1.3 The developer shall select, tailor, and use those standards, methods, tools, and computer programming languages (if not stipulated in the contract) that are documented, appropriate, and established by the organization for

performing the activities of the Development Process and supporting processes (section 6).

5.3.1.4 The developer shall develop plans for conducting the activities of the development process These plans shall be documented and executed. The SDP shall address aspects like development schedule, specific methods, tools and techniques, responsibility associated with the development and qualification of all requirements. **If necessary, separate plans may be developed. SDP can also be a part of SMP.**

5.3.1.5 Non-deliverable items may be employed in the development or maintenance of the software product. However, it shall be ensured that the operation and maintenance of the deliverable software product after its delivery to the acquirer are independent of such items, otherwise those items should be considered as deliverable.

Expected Output: ***Software Development Plan(SDP)***

5.3.2 System Requirements Analysis

Required Input: ***System Requirements/User Requirements***

5.3.2.1 The Developer shall analyse and document the System requirements such as functions and capabilities of the System, User requirements (safety, security, operation and maintenance requirements), Design constraints and qualification requirements. Each requirement shall be stated in such a way that test objectives can be defined for each. Interfaces (both internal and external) shall also be defined in an Interface Requirements document (IRD), which can be a part of System Requirement document or a separate one.

5.3.2.2 The systems requirements shall be evaluated considering the following:

a) Traceability and consistency to acquisition needs
b) Testability and feasibility of system architectural design
c) Feasibility of operation and maintenance.

Expected Output: ***System Requirements Document (SRD)***

5.3.3 System Architectural Design

Required Input: ***System Requirements Document(SRD)***

5.3.3.1 A top-level architecture of the system identifying items of hardware and software shall be carried out. It shall be ensured that all the system requirements are allocated among the items and documented.

5.3.3.2 The system architecture and the requirements for the items shall be evaluated considering the following:

a. Traceability and consistency to system requirements
b. Appropriateness of design standard and methods used
c. Feasibility of software items fulfilling their allocated requirements
d. Feasibility of operation and maintenance.

Expected Output: ***System Architectural Design Document***

5.3.4 Software Requirements Analysis

Required Input: ***System Requirements Document (SRD)***
System Architectural Design Document
User Requirements Document

5.3.4.1 The Developer shall translate the software requirements in compliance with System Requirements Document/System Architecture Design/ User requirements. The requirements specification for each component shall include the following:

a. Functional/capability requirements including performance requirements and the environment conditions under which the software item is to perform
b. Interface requirements (internal and external)
c. Qualification requirements
d. Safety specifications
e. Security specifications
f. Human factors engineering including those related to manual operations, human equipment interactions, constraints on personnel and areas

needing concentrated human attention that are sensitive to human errors and training.

g. Data definition and database requirements
h. Installation and acceptance requirements
i. User documentation
j. User operation and execution requirements
k. User maintenance requirements

The requirements shall be documented in Software Requirements Specification (SRS).

5.3.4.2 The software requirements shall be analysed based on the following:

a. Requirements traceability and consistency with respect to SRD/system design.
b. Interface consistency (external /internal)
c. Testability and feasibility of software design
d. Feasibility of operation and maintenance

5.3.4.3 The developer shall conduct joint review(s) in accordance with the Joint review process(6.6). Upon successful completion, a baseline for the requirements of the software shall be established.

Expected Output: ***Software Requirements Specification (SRS)***

5.3.5 Software Architectural Design

Required Input: ***System Requirements Document (SRD)***
Software Requirements Specification (SRS)

5.3.5.1 The Developer shall transform the requirements for the software item into an architecture that describes its top-level structure (architectural design) and identify the software components and units. It shall be ensured that all the requirements for the software item defined in SRS are allocated to its software components. These may further be refined to facilitate detailed design.

5.3.5.2 A top-level design for the interfaces external to the software and between the components of the software, database (if any) shall be carried out.

5.3.5.3 The developer shall develop and document a top-level design for the database.

5.3.5.4 The developer should define and document preliminary versions of user documentation.

5.3.5.5 The developer shall define and document preliminary test requirements and the schedule for Software Integration.

5.3.5.6 The developer shall evaluate the architecture of the software item and the interface and designs based on the following criteria . The evaluations shall be documented.

a. Traceability of requirements
b. External /Internal consistency
c. Appropriateness of design methods, process documents
d. Feasibility of detailed design
e. Feasibility of operation and maintenance

5.3.5.7 The developer shall conduct joint review(s) in accordance with the Joint review process 6.6.

Expected Output: ***Software Architectural Design Document (SAD)***

5.3.6 Software Detailed Design

Required Input: ***Software Requirements Specification(SRS)***
Software Architectural Design Document(SAD)

5.3.6.1 The developer shall develop a detailed design for each software component of the software item. The software components shall be refined into lower levels containing software units that can be coded, compiled and tested. It shall be ensured that all the software requirements are allocated from the software components to software units and the detailed design shall be documented.

5.3.6.2 The developer shall carry out a detailed design for the interfaces external to the software and between the software components and between the units.

The detailed design of the interfaces shall permit coding without the need for further information.

5.3.6.3 For database design (if applicable), the designer shall develop a detailed design for the database.

5.3.6.4 The developer shall update user documentation as necessary.

5.3.6.5 The designer shall define and document test requirements and schedule for testing software units.

5.3.6.6 The developer shall update the test requirements and schedule for Software Integration.

5.3.6.7 The developer shall evaluate the design based on the following criteria and the results of the evaluations shall be documented.

a. Traceability of design to requirements
b. External /Internal consistency
c. Appropriateness of design methods and standards used
d. Feasibility of testing
e. Feasibility of operation and maintenance

5.3.6.8 The designer shall conduct joint review(s) in accordance with the Joint review process.

Expected Output: ***Software Design Document (SDD)***
User Manual (if applicable)

5.3.7 Software Coding and Testing

Required Input: ***Software Design Document (SDD)***

5.3.7.1 The detailed design of the software components and units shall be coded in the desired language. The test procedures and data for testing each software and database shall be worked out.

5.3.7.2 The developer shall design test cases for testing software units and carry out unit testing to ensure correctness, traceability and external consistency to

requirements and design of the software, test coverage of units, internal consistency between the software units etc.

5.3.7.3 The developer shall update the user manual as necessary.

5.3.7.4 The developer shall update the test requirements and the schedule for Software integration.

5.3.7.5 The developer shall evaluate software code and test results considering the following and the evaluations shall be documented.

a. Traceability of code to design and requirements
b. External /Internal consistency
c. Test coverage of units
d. Appropriateness of coding methods and standards used
e. Feasibility of software integration and testing
f. Feasibility of operation and maintenance

Expected Output: ***Source code***
Test Results

5.3.8 Software Integration

Required Input: ***Source Code***
Software Requirements Specification(SRS)

5.3.8.1 The Integration engineer shall develop an Integration Plan to integrate the software units into the integrated software. This plan shall include test requirements, procedures, data, responsibilities and schedule.

5.3.8.2 The Integration engineer shall integrate the software units and shall test that the aggregates are developed in accordance with the Integration Plan. It shall be ensured that each aggregate satisfies the requirements of the software and that the software is integrated at the end of the integration activity. The data associated with the individual components of the software shall also be integrated.

5.3.8.3 The user manual may be updated as a result of Integration testing.

5.3.8.4 The developer shall develop and document, for each qualification requirement of the software item, a set of tests, test cases (inputs, outputs, test criteria), and test procedures for conducting Software Qualification Testing. The developer shall ensure that the integrated software item is ready for Software Qualification Testing.

5.3.8.5 The Integration Engineer shall evaluate the plan, design, code, tests and test results considering the following:

a. Traceability to software and system requirements
b. External consistency with system requirements
c. Internal consistency between the software units
d. Test coverage of the requirements of the software
e. Appropriateness of test standards and methods used
f. Conformance to expected results
g. Feasibility of Software Qualification testing
h. Feasibility of operation and maintenance

5.3.8.6 The developer shall conduct joint review(s) in accordance with 6.6.

Expected Output: ***Software Integration Plan***
Software Integration Test Results
Integrated Software
Software Qualification Test Plan
User manual (if applicable)

5.3.9 Software Qualification Testing

Required Input: ***Software Qualification Test Plan***
Integrated Software
User Manual (if applicable)

5.3.9.1 The developer shall conduct qualification testing in accordance with the qualification requirements for the software item. It shall be ensured that the implementation of each software requirement is tested for compliance. The qualification testing results shall be documented.

5.3.9.2 The developer shall update the user documentation as necessary.

5.3.9.3 The designer shall evaluate the design, code, tests and test results considering the following:

a. Test coverage of the requirements of the software
b. Conformance to expected results
c. Feasibility of system integration and testing (if applicable)
d. Feasibility of operation and maintenance

5.3.9.4 The designer shall support audit as per the 'Audit process'. If both hardware and software are under development or integration, the audits may be postponed until System Qualification Testing.

5.3.9.5 Upon successful completion of audits, if conducted, the designer shall

a. Update and prepare the deliverable software product for System Integration, System Qualification Testing, Software Installation or Software Acceptance support as applicable.

b. Establish a baseline for design and code of the software

Note: The Software Qualification testing may be used in the Verification Process or Validation process.

Expected Output: ***Software Qualification Test Results***
Audit report (if applicable)

5.3.10 System Integration

Required Input: ***Integrated Software***
System Requirements Specification
Target Hardware

5.3.10.1 The System Integration engineer shall integrate software configuration item(s) with hardware configuration items, manual operations and other systems as necessary into the system. The aggregates shall be verified/ tested as they are developed, against their requirements. The results shall be documented.

5.3.10.2 For each qualification requirement of the system, a set of tests, test cases (inputs, outputs, test criteria), and test procedures for conducting System Qualification Testing shall be developed and documented. The developer shall ensure that the integrated system is ready for System Qualification Testing.

5.3.10.3 The integrated system shall be evaluated considering the following:

a. Test coverage of system requirements
b. Appropriateness of test methods and standards used.
c. Conformance to expected results
d. Feasibility of system qualification testing
e. Feasibility of operation and maintenance

Expected Output: ***Integrated System***
Evaluation reports
System Qualification Test Plan

5.3.11 System Qualification Testing

Required Input: ***System Qualification Test Plan***
Integrated System
User Manual

5.3.11.1 System qualification testing shall be conducted in accordance with the qualification requirements specified for the system. It shall be ensured that the implementation of each system requirement is tested for compliance and that the system is ready for delivery. The qualification testing results shall be documented.

5.3.11.2 The system shall be evaluated based on the following:

a. Test coverage of system requirements
b. Conformance to expected results
c. Feasibility of operation and maintenance

5.3.11.3 The designer shall support audit as per the 'Audit process'(6.7).

5.3.11.4 Upon successful completion of audits, if conducted, the designer shall

a. Update and prepare the deliverable software product for Software Installation or Software Acceptance support.
b. Establish a baseline for design and code of each software configuration item.

Expected Output: ***System Qualification Test Results***
Audit report (if applicable)

Note:

i. The System Qualification testing may be used in the Verification Process or Validation process.
ii. If audits were conducted previously for software configuration items (5.3.9), it need not be carried out again.

5.3.12 Software Installation

Required Input: ***Integrated Software***

5.3.12.1 An installation plan shall be generated to install the software in the target environment. The plan shall include the resources and information necessary to install the software.

5.3.12.2 The software shall be installed in the target platform in accordance with the installation plan. The installation events and results shall be documented.

Expected Output: ***Installation Plan***
Installed software

5.3.13 Software Acceptance Support

Required Input: ***Integrated Software***
Verification reports
Test results

5.3.13.1 The designer shall support the users acceptance review and testing of the software product. Acceptance review and testing shall consider the results of Joint Reviews, Software Qualification Testing and System Qualification Testing (if applicable). The review and test results shall be documented.

5.3.13.2 The designer shall complete and deliver the software product and provide initial and continuing training and support to the users.

5.3.13.3 The developer shall provide initial and continuing training and support to the acquirer as specified in the contract.

Expected Output: ***Deliverable software***

5.4 Operation Process

Entry Criteria: ***Completion of Development, Validation, Acceptance & installation***

The operation process contains the activities and tasks of the operator. The process covers the operation of the software and operational support to the users.

This process consists of the following activities:

- **Operation planning**
- **Operational Testing**
- **System Operation**
- **User Support**

Exit Criteria: ***Software Retirement /Project Closure***

5.4.1 Operation Planning

Required Input: ***User Manual***

5.4.1.1 The operator shall develop an Operation plan for performing the operational activities and set operational standards for performing the activities and tasks of this process. The plan shall be documented and executed.

5.4.1.2 The operator shall establish procedures for receiving, recording, resolving, tracking problems, and providing feedback. Whenever problems are encountered, they shall be recorded and entered into the Problem Resolution Process (6.8).

5.4.1.3 The operator shall establish procedures for testing the software product in its operation environment, for entering problem reports and modification

requests to the Maintenance Process (5.5) and for releasing the software product for operational use.

Expected Output: ***Operation Plan***
Operation Manual / Procedures

5.4.2 Operational Testing

Required Input: ***Operation Plan***

5.4.2.1 The operator shall perform operational testing to ensure that the software executes as defined in the Operation plan. The software shall be released for system operation, if it passes the operational test.

5.4.2.2 The operator shall ensure that the software code and databases initialize, execute, and terminate as described in the plan.

Expected Output: ***Operation Test results***
Release of Operational software

5.4.3 System Operation

Required Input: ***Operation Manual / Procedures***

5.4.3.1 The system shall be operated in its intended environment according to the user documentation.

Expected Output: ***Operation Log***

5.4.4 User Support

Required Input: ***User Requests***

5.4.4.1 The operator shall provide necessary assistance and consultation to the users of the system as requested. These requests and subsequent actions shall be recorded and monitored.

5.4.4.2 The operator shall forward user requests as necessary to the Maintenance process for resolution. These requests shall be addressed and the actions

that are planned and taken shall be reported to the originators of the requests. All resolutions shall be monitored to conclusion.

5.4.4.3 If a reported problem has a temporary work-around before a permanent solution can be released, the originator of the problem report shall be given the option to use it. Permanent corrections, releases that include previously omitted functions or features, and system improvements shall be applied to the operational software product using the Maintenance Process (5.5).

Expected Output: ***User Support Log***
Problem resolutions

5.5 Maintenance Process

The maintenance process is initiated when the software undergoes modification due to a problem, requirement change, optimization or adaptation. This process contains the activities and tasks related to the software changes, problem and modification analysis, change implementation, change review/ acceptance of the software. This process may address migration and retirement of software, if applicable.

Entry Criteria: ***Need for change/SMP***

The following activities are addressed in this process

- **Maintenance Planning**
- **Problem and Modification analysis**
- **Modification Implementation**
- **Maintenance Review / Acceptance**
- **Migration**
- **Software Retirement**

Exit Criteria: ***Change Implementation /Project Closure***

5.5.1 Maintenance Planning

Required Input: ***Need for change***

5.5.1.1 The Maintenance Plan shall be developed describing the activities, which are to be carried out during the maintenance phase.

5.5.1.2 The Plan shall address techniques for receiving, recording and tracking problem reports and change requests from the users. The plan shall also address activities related to problem/ modification analysis, impact of change implementation and its review/acceptance.

5.5.1.3 The maintainer shall implement (or establish organizational interface with) the Configuration Management Process (6.2) for managing modifications to the existing system.

Expected Output: ***Software Maintenance Plan (SMNP)***

5.5.2 Problem and modification analysis

Required Input: ***Problem report***

5.5.2.1 The maintainer shall analyze the problem report or modification request for its impact on the organization, the existing system, and the interfacing systems.

5.5.2.2 The maintainer should replicate or verify the problem.

5.5.2.3 Based upon the analysis, the maintainer shall consider options for implementing the modification.

5.5.2.4 The maintainer shall document the problem/modification request, the analysis results, and implementation options.

5.5.2.5 The maintainer shall obtain approval for the selected modification option as specified in the contract.

Expected Output: ***Analysis Report***
Software change requests

5.5.3 Modification Implementation

Required Input: ***Software Change Requests***

5.5.3.1 The maintainer shall conduct analysis and determine which documentation, software units, and versions thereof need to be modified. These shall be documented.

5.5.3.2 For the implementation of the approved changes, the maintainer shall enter the appropriate phase in the software Development Process (5.3). The requirements of the Development Process shall be supplemented as follows:

a) Test and evaluation criteria for testing and evaluating the modified and the unmodified parts (software units, components, and configuration items) of the system shall be defined and documented.

b) The complete and correct implementation of the new and modified requirements shall be ensured. It also shall be ensured that the

original, unmodified requirements were not affected. The test results shall be documented.

Expected Output: ***Modified software***

5.5.4 Maintenance Review/Acceptance

Required Input: ***Modified software***

5.5.4.1 The maintainer shall conduct review(s) with the organization authorizing the modification to determine the integrity of the modified system.

5.5.4.2 The maintainer shall obtain approval for the satisfactory completion of the modification as specified in the contract.

Expected Output: ***Review report***

5.5.5 Migration

Required Input: ***Migration Requirements***

5.5.5.1 If a system or software product (including data) is migrated from an old to a new operational environment, it shall be ensured that any software product or data produced or modified during migration are in accordance with ISPD.

5.5.5.2 A migration plan shall be generated addressing the following activities:

a) Requirement analysis and definition of migration
b) Development of migration tools
c) Conversion of software product and data
d) Migration execution and verification
e) Support for the old environment in the future

5.5.5.3 Users shall be given notification of the migration plans and activities. Notifications shall include the following:

a) Statement of why the old environment is no longer to be supported;
b) Description of the new environment with its date of availability;

c) Description of other support options available, if any, once support for the old environment has been removed.

5.5.5.4 Parallel operations of the old and new environments may be conducted for smooth transition to the new environment.

5.5.5.5 When the scheduled migration arrives, notification shall be sent to all concerned. All associated old environment's documentation, logs, and code should be placed in archives.

5.5.5.6 A post-operation review shall be performed to assess the impact of changing to the new environment.

5.5.5.7 Data used by or associated with the old environment shall be accessible in accordance with the contract requirements for data protection and audit applicable to the data.

Expected Output: ***Migration plan***
Migrated Software / System

5.5.6 Software Retirement

Required Input: ***Retirement Request***

5.5.6.1 A retirement plan shall be generated addressing the following:

a) Cessation of full/partial support after a certain period of time
b) Archiving of software product and associated documents
c) Responsibility for any future residual support issues
d) Transition to new software product if applicable
e) Accessibility of archived copies of data

5.5.6.2 The users shall be given notification of the retirement plans and activities. Notifications shall include the following:

a) Description of the replacement or upgrade with its date of availability
b) Statement of why the software product is no longer to be supported

c) Description of other support options available, once support has been removed.

5.5.6.3 Parallel operations of the retiring and the new software product should be conducted for smooth transition to the new system. During this period, user training shall be provided as specified in the contract.

5.5.6.4 When the scheduled retirement arrives, notification shall be sent to all concerned. All associated development documentation, logs, and code should be placed in archives, when appropriate. The experiences on the software project shall be documented for future purpose.

5.5.6.5 Data used or associated by the retired software product shall be accessible in accordance with the contract requirements for data protection and audit applicable to the data.

Expected Output: ***Retirement plan/Archives***

6

Supporting Life Cycle Processes

The following supporting life cycle processes are defined:

- **Documentation Process**
- **Configuration Management Process**
- **Quality Assurance Process**
- **Verification Process**
- **Validation Process**
- **Joint Review Process**
- **Audit Process**
- **Problem Resolution Process**

6.1 Documentation Process

This is a process for recording information produced by a life cycle process or activity. This contains the set of activities, which addresses the generation, editing, distribution, maintenance of documents and records pertaining to software product.

This consists of the following activities

- **Documentation Planning**
- **Design and Development**
- **Production**
- **Maintenance**

6.1.1. Documentation Planning

Required Input: ***Software Project Management Plan (SMP)***

6.1.1.1 A plan identifying the documents to be produced during the life cycle of the software product and the responsibility shall be generated as per this process document. For each identified document, the following shall be addressed:

a) Title or name;
b) Purpose;
c) Intended audience;
d) Procedures and responsibilities for inputs, development, review, modification, approval, production, storage, distribution, maintenance, and configuration management;
e) Schedule for intermediate and final versions.

Expected Output: ***Documentation Plan / Procedure***

6.1.2 Design and Development

Required Input: ***Documentation Plan / Procedure***

6.1.2.1 The document shall be designed in accordance with the applicable documentation standards for format, content description, page numbering etc.

6.1.2.2 The source and appropriateness of input data for the documents shall be confirmed. Automated documentation tools may be used.

6.1.2.3 These documents shall be reviewed and audited as per the Joint review and Audit processes (6.6 and 6.7) respectively.

Required Input: ***Reviewed Document***

6.1.3 Production

Required Input: ***Reviewed Document***

6.1.3.1 The documents shall be produced and provided in accordance with the applicable documentation standards. Production and distribution of documents may use paper, electronic or other media. Master materials shall be stored in accordance with the requirements for record retention, security, maintenance and backup.

6.1.3.2 Control of documents shall be carried out in accordance with a documented procedure.

Expected Output: ***Software Documents (Eg: SRS, SDD,Plans)***

6.1.4 Maintenance

Required Input: ***Document modifications***

6.1.4.1 For the documents that are under configuration management, modifications are to be implemented as per the Configuration Management Process (6.2)

Expected Output: Modified Document

6.2 Configuration Management (CM) Process

This process is applied throughout the software life cycle. This process addresses the identification, definition and base lining of software item, version control and release of the software product. A designated person/team shall carry out the CM process.

The following activities are addressed in this process:

- **Configuration Management Planning**
- **Configuration Identification**
- **Configuration Control**
- **Configuration Status Accounting**
- **Configuration Evaluation**
- **Release Management and delivery**

6.2.1 Configuration Management Planning

Required Input: ***SMP***

6.2.1.1 The concerned designer/CM Team shall develop a CM Plan. The Plan shall describe the Configuration Management activities, procedures, schedule, version numbering scheme, tools used (if any) for performing the activities and responsibilities.

Expected Output: ***Configuration Management Plan***

6.2.2 Configuration Identification

Required Input: ***CM Plan***
Software Items

6.2.2.1 A scheme shall be established for the identification of the software configuration items and their version control. For each software configuration item and its versions, documentation that establishes the baseline, version references and identification details shall be provided.

Note: Data associated with the software shall also be identified separately and version controlled (if applicable).

Expected Output: ***Configuration Item***

6.2.3 Configuration Control

Required Input: ***Configuration Item***

6.2.3.1 The following shall be performed: identification and recording of change requests; analysis and evaluation of the changes; approval or disapproval of the request; and implementation, verification, and release of the modified software item. An audit trail shall exist, whereby each modification, the reason for the modification, and authorization of the modification can be traced. Control and audit of all accesses to the controlled software items that handle safety or security critical functions shall be performed.

Expected Output: ***Configuration Controlled Item***

6.2.4 Configuration Status Accounting

Required Input: ***Configuration Item***

6.2.4.1 The activities include preparation of management records and status reports, that show baseline and change history, latest software versions, release identifiers, number of releases, comparison of releases etc.

Expected Output: ***CM Reports***

6.2.5 Configuration Evaluation

Required Input: ***Configuration Item***

6.2.5.1 The functional completeness of the software against their requirements and physical completeness of software items whether design and code reflect up to date technical description shall be determined and ensured.

Expected Output: ***Configuration Evaluation Report***

6.2.6 Release Management and Delivery

Required Input: ***Configuration Item***

6.2.6.1 The release and delivery of software products and documentation shall be formally controlled. Master copies of software code and documentation shall be maintained for the life of the software product.

Expected Output: ***Controlled Software code / Document***

6.3 Software Quality Assurance (SQA) Process

The objective of this process is to provide adequate assurance that the software process and product conform to the specified requirements and adhere to established plans and process documents. The following activities are addressed in this process:

- **SQA Planning**
- **Process assurance**
- **Product assurance**
- **Assurance of quality systems**

This process may use the results of other supporting processes such as Joint review, Audit, Verification, Validation and Problem resolution to satisfy the quality assurance function.

6.3.1 SQA Planning

Required Input: ***Software Project Management Plan (SMP)***

6.3.1.1 A quality assurance process tailored to the project shall be established. The objectives of the quality assurance process shall be to assure that the software products and the processes employed for providing those software products comply with their established requirements and adhere to their established plans.

6.3.1.2 The quality assurance process should be coordinated with the related Verification (6.4), Validation (6.5), Joint Review (6.6), and Audit (6.7) Processes

6.3.1.3 A Plan shall be generated for conducting the QA activities and tasks. The Plan shall include the following:

a) Quality process documents, methodologies, procedures
b) Responsibilities and tools for performing the QA tasks
c) Schedules and milestones
d) Reviews and problem resolution / risk analysis

e) V&V activities
f) Quality audits
g) Identification and maintenance of quality records

6.3.1.4 Scheduled and on-going quality assurance activities and tasks shall be executed. When problems or nonconformance with contract requirements are detected, they shall be documented and serve as input to the Problem Resolution Process (6.8). Records of these activities and tasks, their execution, problems, and problem resolutions shall be prepared and maintained.

6.3.1.5 Records of quality assurance activities and tasks shall be made available to the acquirer as specified in the contract.

6.3.1.6 It shall be assured that persons responsible for assuring compliance with the contract requirements have the organizational freedom, resources, and authority to permit objective evaluations and to initiate, effect, resolve, and verify problem resolutions.

Expected Output*:** ***Software Quality Assurance Plan (SQAP)

6.3.2 Product Assurance

Required Input: ***Software Quality Assurance Plan (SQAP)***
Software products

6.3.2.1 The Product assurance shall ensure that all products specified in the defined plans exist.

6.3.2.2 It shall be ensured that the product and related documentation adhere to the defined process documents and have undergone evaluation as defined in the plans.

6.3.2.3 In preparation for the delivery of the software products, it shall be assured that they have fully satisfied their contractual requirements and are acceptable to the acquirer.

This process can use the results of verification, validation, joint reviews and audits.

Expected Output: ***Product Assurance reports***

6.3.3 Process Assurance

Required Input: ***Software Quality Assurance Plan (SQAP)***
Software Project Management Plan(SMP)

6.3.3.1 The Process assurance shall be assured that the process shall be adhered to the software life cycle processes as defined in the plans.

6.3.3.2 It shall be assured that the internal software engineering practices, development environment, test environment, and libraries comply with the contract.

6.3.3.3 It shall be assured that applicable prime-contract requirements are pass down to the subcontractor, and that the sub-contractor's software products satisfy prime contract requirements.

6.3.3.4 It shall be assured that the acquirer and other parties are provided the required support and cooperation in accordance with the contract negotiations and plans.

6.3.3.5 It shall be assured that the product and process measurements are in accordance with the established process documents.

6.3.3.6 It shall be assured that the staff assigned has the skill needed to meet the requirements of the project.

Expected Output: ***Process Assurance reports***

6.3.4 Assurance of Quality Systems

Required Input: ***QMS document,***
SMP

6.3.4.1 Additional quality management activities shall be assured to comply with clauses of ISO 9001 through Internal Quality Audits.

Expected Output: ***Internal Audit Reports***

6.4 Verification Process

This process determines whether the software product actually fulfills the requirements intended in the previous phases. The following activities are addressed in this process:

- **Verification Planning**
- **Verification**

Results and observations of these activities shall be documented and reviewed. If an anomaly is observed, it shall be communicated to Developer for corrective action.

Note: For mission critical software, this process shall be executed by an independent agency. For all other types of software, this can be executed either by an independent agency or by a team specially constituted for the purpose.

6.4.1 Verification Planning

Required Input: ***Software Quality Assurance Plan (SQAP)***

6. 4.1.1 A determination shall be made if the project warrants a verification effort and the degree of organizational independence of that effort needed. The project requirements shall be analyzed for criticality. Criticality may be gauged in terms of:

a) The potential of an undetected error in a system or software requirement for causing death or personal injury, mission failure, or financial or catastrophic equipment loss or damage;
b) The maturity of and risks associated with the software technology to be used
c) Availability of funds and resources.

6.4.1.2 If the project warrants a verification effort, a verification process shall be established to verify the software product.

6.4.1.3 If the project warrants an independent verification effort, a qualified organization responsible for conducting the verification shall be selected. This organization shall be assured of the independence and authority to perform the verification activities.

6.4.1.4 Based upon the scope, magnitude, complexity, and criticality analysis above, target life cycle activities and software products requiring verification shall be determined. Verification activities and tasks defined in 6.4.2, including associated methods, techniques, and tools for performing the tasks, shall be selected for the target life cycle activities and software products.

6.4.1.5 Based upon the verification tasks as determined, a Plan for carrying out the Verification activities at various phases shall be generated.

The plan shall address the following aspects:

a) Methods and criteria, and inputs/outputs
b) Schedules, milestones and resources
c) Problem reporting / resolution
d) Task iteration policy, waiver / deviation policy
e) Process documents, practices and conventions

6.4.1.6 The verification plan shall be implemented. Problems and non conformances detected by the verification effort shall be entered into the Problem Resolution Process (6.8). All problems and non conformances shall be resolved. Results of the verification activities shall be made available to the acquirer and other involved organizations.

Expected Output: ***Software Verification Plan/Software V&V Plan***

6.4.2 Verification

6.4.2.1 Contract Verification

Required Input: ***Contract***

The contract shall be verified for the following criteria:

a) The supplier has the capability to satisfy the requirements.
b) The requirements are consistent and cover user needs
c) Adequate procedure for handling changes to requirements and escalating problems are stipulated
d) Procedures and their extent for interface and cooperation among the parties are stipulated, including ownership, warranty, copyright and confidentiality.
e) Acceptance criteria and procedures are stipulated in accordance with requirements.

Expected Output: ***Reviewed contract***

6.4.2.2 Process Verification

Required Input: ***Software Project Management Plan (SMP)***

The process shall be verified for the following criteria:

a) The project planning requirements are adequate and timely.
b) Processes selected for the project are adequate, implemented, being executed as planned and compliant with the contract.
c) The standards, procedures and environment for the projects processes are adequate.
d) The project is staffed and personnel trained as required by the contract.

Expected Output: ***Process verification Report***

6.4.2.3 Requirements Verification

Required Input: ***System Requirements Document (SRD)***
Software Requirements Specification (SRS)
Software Verification & Validation Plan (SVVP)

The requirements shall be verified to ensure the following:

a) The system requirements are consistent, feasible, and testable.
b) The system requirements have been appropriately allocated to hardware items, software items, and manual operations according to design criteria.
c) The software requirements are consistent, feasible, testable, and accurately reflect system requirements.
d) The software requirements related to safety, security, and criticality are correct as shown by suitably rigorous methods.

Expected Output: ***Requirements Verification report***

6.4.2.4 Design Verification

Required Input: ***Software Verification & Validation Plan(SVVP)***
Software Design Document(SDD)

The design shall be verified considering the following:

a) The design is correct and consistent with and traceable to requirements.
b) The design implements proper sequence of events, inputs, outputs, interfaces, logic flow, allocation of timing and sizing budgets, and error definition, isolation, and recovery.
c) Selected design can be derived from requirements.
d) The design implements safety, security, and other critical requirements correctly as shown by suitably rigorous methods.

Expected Output: ***Design Verification report***

6.4.2.5 Code Verification

Required Input: ***Software Verification & Validation Plan (SVVP)***
Configuration Controlled Code
Software Detailed Design (SDD)
Software Requirements Specification (SRS)

Based on the criticality of the software, the following verification tasks may be identified and stated in SVVP.

a) The code is traceable to design and requirements, testable, correct, and compliant with requirements and coding standards.
b) The code implements proper event sequence, consistent interfaces, correct data and control flow, completeness, appropriate allocation timing and sizing budgets, and error definition, isolation, and recovery.
c) Selected code can be derived from design or requirements.
d) The code implements safety, security, and other critical requirements correctly as shown by suitably rigorous methods.

Expected Output: ***Code Verification report***

6.4.2.6 Integration Verification

Required Input: ***Integrated Software***
Integration Plan

The integration shall be verified considering the criteria listed below:

a. The software components and units of each software item have been completely and correctly integrated into the software item.
b. The hardware items, software items, and manual operations of the system have been completely and correctly integrated into the system.

c. The integration tasks have been performed in accordance with an integration plan.

Expected Output: ***Integration Verification report***

6.4.2.7 Documentation verification

Required Input: ***Software Documents***

The documentation shall be verified considering the criteria listed below:

a. The documentation is adequate, complete, and consistent.
b. Documentation preparation is timely.
c. Configuration management of documents follows specified procedures.

Expected Output: ***Documentation Verification report***

6. 5. Validation Process

This process determines whether the requirements and the final, as-built system or software product fulfills its specific intended use. This process may be conducted as a part of Software Acceptance Support (5.3.13). This process may be executed with varying degrees of independence. The degree of independence may range from the same person or different person in the same organization to a person in a different organization with varying degrees of separation. In the case where the process is executed by an organization independent of the supplier, developer, operator, or maintainer, it is called Independent Validation Process.

The following activities are addressed in this process:

- **Validation Planning**
- **Validation**

Note: For mission critical software, this process shall be executed by an independent agency. For all other types of software, this can be executed either by an independent agency or by a team specially constituted for the purpose.

6. 5.1. Validation Planning

Required Input: ***System Requirements Specification (SRD)***
Software Requirements Specification (SRS)

6.5.1.1 A determination shall be made if the project warrants a validation effort and the degree of organizational independence of that effort needed.

6.5.1.2 If the project warrants a validation effort, a validation process shall be established to validate the system or software product. Validation tasks defined below, including associated methods, techniques, and tools for performing the tasks, shall be selected.

6.5.1.3 If the project warrants an independent effort, a qualified organization responsible for conducting the effort shall be selected. The conductor shall be assured of the independence and authority to perform the validation tasks.

6.5.1.4 A Validation Plan (VP) shall be generated by the identified agency in the V&V plan. The plan shall address the following:

a) Items subject to validation

b) Validation tasks to be performed

c) Resources, responsibilities, and schedule for validation

d) Procedures for forwarding validation reports to the acquirer and other parties.

6.5.1.5 The validation plan shall be implemented. Problems and non-conformances detected by the validation effort shall be entered into the Problem Resolution Process (6.8). All problems and non-conformances shall be resolved. Results of the validation activities shall be made available to the acquirer and other involved organizations.

Expected Output: ***Validation Plan/Software V&V Plan (SVVP)***

6.5.2 Validation

Required Input: ***Validation Plan/Software V&V Plan (SVVP)***
System Requirements Specification/Software Requirements Specification (SRS)

6.5.2.1 Prepare selected test requirements, test cases, and test specifications for analyzing test results.

6.5.2.2 Ensure that these test requirements, test cases, and test specifications reflect the particular requirements for the specific intended use.

6.5.2.3 Conduct the tests in subclauses 6.5.2.1 and 6.5.2.2, including:

a) Testing with stress, boundary, and singular inputs;

b) Testing the software product for its ability to isolate and minimize the effect of errors; that is, graceful degradation upon failure, request for operator assistance upon stress, boundary, and singular conditions;

c) Testing that representative users can successfully achieve their intended tasks using the software product.

6.5.2.4 Validate that the software product satisfies its intended use.

6.5.2.5 Test the software product as appropriate in selected areas of the target environment.

Expected Output: ***Validation Test Results***
Test case document

6.6. Joint Review Process

This process addresses the activities for evaluating the status and products of various phases in software life cycle. Reviews are held both at Management and technical levels. This process consists of the following activities.

- **Review Planning**
- **Project Management Reviews**
- **Technical Reviews**

6.6.1 Review Planning

Required Input: ***Software Project Management Plan (SMP)***

6.6.1.1 Periodic reviews shall be held at pre-determined milestones as specified in the SMP.

6.6.1.2 All resources required to conduct the reviews shall be agreed on by the parties. These resources include personnel, location, facilities, hardware, software, and tools.

6.6.1.3 The parties should agree on the following items at each review: meeting agenda, software products (results of an activity) and problems to be reviewed; scope and procedures; and entry and exit criteria for the review.

6.6.1.4 Problems detected during the reviews shall be recorded and entered into the Problem resolution process (8.8) as required.

6.6.1.5 The review results shall be documented and distributed. The reviewing party will acknowledge to the reviewed party the adequacy (for example, approval, disapproval, or contingent approval) of the review results.

6.6.1.6 The parties shall agree on the outcome of the review and any action item responsibilities and closure criteria.

Expected Output: ***Review plan***

6.6.2 Project Management Reviews

Required Input: ***Review Document / Plans***

6.6.2.1 Project status shall be evaluated relative to the applicable project plans, schedules, standards, and guidelines. The outcome of the review should be discussed between the two parties and should provide for the following:

a) Making activities progress according to plan, based on an evaluation of the activity or software product status
b) Maintaining global control of the project through adequate allocation of resources
c) Changing project direction or determining the need for alternate planning
d) Evaluating and managing the risk issues that may jeopardize the success of the project

Expected Output: ***Review records***

6.6.3 Technical Reviews

Required Input: ***Review Document / Plans***

6.6.3.1 Technical reviews shall be held to evaluate software products or services under consideration to ensure that

a) They are complete.
b) They comply with their standards and specifications.
c) Changes to them are properly implemented and affect only those areas identified by the Configuration Management Process (6.2).
d) They are adhering to applicable schedules.
e) They are ready for the next planned activity.
f) The development, operation, or maintenance is being conducted according to the plans, schedules.

Expected Output: ***Review records***

6.7 Audit Process

This is a process for determining the compliance with requirements, plans and process documents.

The following activities are addressed in this process:

- **Audit Planning**
- **Audit.**

6.7.1 Audit planning

Required Input: ***Software Project Management Plan (SMP)***

6.7.1.1 Audits shall be held at predetermined milestones as specified in the project plan(s).

6.7.1.2 Auditing personnel shall not have any direct responsibility for the software products and activities they audit.

6.7.1.3 All resources required to conduct the audits shall be agreed by the parties. These resources include supporting personnel, location, facilities, hardware, software, and tools.

6.7.1.4 The parties should agree on the following items at each audit: agenda; software products (and results of an activity) to be reviewed; audit scope and procedures; and entry and exit criteria for the audit.

6.7.1.5 Problems detected during the audits shall be recorded and entered into the Problem Resolution Process (6.8) as required.

6.7.1.6 After completing an audit, the audit results shall be documented and provided to the audited party. The audited party shall acknowledge to the auditing party any problems found in the audit and related problem resolutions planned.

6.7.1.7 The parties shall agree on the outcome of the audit and any action item responsibilities and closure criteria.

Expected Output: ***Audit Plan***

6.7.2 Audit

Required Input: ***Software Products***

6.7.2.1 Audits shall be held to ensure the following

a) Correct life cycle model is followed, as defined in SMP
b) Prescribed process documents/guidelines are adhered to
c) As-coded software products (such as a software item) reflect the design documentation.
d) The acceptance review and testing requirements prescribed by the documentation are adequate for the acceptance of the software products.
e) Test data comply with the specification.
f) Software products were successfully tested and meet their specifications.
g) Test reports are correct and discrepancies between actual and expected results have been resolved.
h) User documentation complies with standards as specified.
i) Activities have been conducted according to applicable requirements, plans, and contract.
j) The costs and schedules adhere to the established plans.

Expected Output: ***Audit Reports***

6.8 Problem Resolution

This process addresses the process for analyzing and resolving the problems that are discovered during the execution of processes such as development, operation, maintenance etc. The objective is to ensure that all discovered problems are analysed, resolved and trends are recognized. This comprises the following activities:

- **Problem resolution planning**
- **Problem resolution**

6.8.1 Problem resolution planning

6.8.1.1 A problem resolution plan shall be generated for handling all problems (including non conformances) detected in the software products and activities. This may be part of the SMP. The plan shall address the following

a) Mechanism for ensuring that: all detected problems are promptly reported and entered into the Problem Resolution Process; action is initiated on them; relevant parties are advised of the existence of the problem as appropriate; causes are identified, analyzed, and, where possible, eliminated; resolution and disposition are achieved; status is tracked and reported; and records of the problems are maintained as stipulated in the contract.

b) A scheme for categorizing and prioritizing the problems. Each problem should be classified by the category and priority to facilitate trend analysis and problem resolution.

c) Analysis shall be performed to detect trends in the problems reported.

d) Problem resolutions and dispositions shall be evaluated: to evaluate that problems have been resolved, adverse trends have been reversed, and changes have been correctly implemented in the appropriate software products and activities; and to determine whether additional problems have been introduced.

Expected Output: ***Problem Resolution Plan***

6.8.2 Problem Resolution

Required Input: ***Product / Process Non-Conformances***

6.8.2.1 When problems (including nonconformance) have been detected in a software product or an activity, a problem report shall be prepared to describe each problem detected. The problem report shall be used as part of the plan described in 6.8.1 from detection of the problem, through investigation, analysis and resolution of the problem and its cause, and onto trend detection across problems.

Expected Output: ***Problem Resolution Report***

7

Organizational Life Cycle Processes

The following are the organizational processes in ISPD

- **Management**
- **Infrastructure**
- **Improvement**
- **Training.**

7.1 Management Process

This process contains the activities and tasks, which may be employed by any agency that has to manage its respective processes. This consists of

- **Initiation and scope definition**
- **Management Planning**
- **Execution and Control**
- **Review and Evaluation**
- **Closure**

7.1.1 Initiation and Scope definition

Required Input: ***Process requirement***

7.1.1.1 The management process shall be initiated by establishing the requirements of the process to be undertaken.

7.1.1.2 Once the requirements are established, the manager shall establish the feasibility of the process by checking that the resources (personnel, materials, technology, and environment) required to execute and manage the process are available, adequate, and appropriate and that the time-scales to completion are achievable.

7.1.1.3 As necessary, and by agreement of all parties concerned, the requirements of the process may be modified at this point to achieve the completion criteria.

Expected Output: ***Process Initiation***

7.1.2 Management Planning

Required Input: ***Software Project Management Plan (SMP)/Process Initiation***

7.1.2.1 The management agency shall identify the plans for execution of various processes. The plans shall contain the descriptions of the associated activities and tasks and identification of the software products that will be provided by the process. These plans shall include, but not limited to, the following:

a) Schedules for the timely completion of tasks
b) Resources needed to execute the tasks
c) Allocation of tasks and responsibilities
d) Assignment of responsibilities
e) Risk identification and quantification
f) Quality control measures
g) Estimation of effort
h) Cost associated with process execution
i) Provision of environment and infrastructure

Expected Output: ***Plans***

7.1.3 Execution and Control

Required Input: ***Software Project Management Plan (SMP)***

7.1.3.1 The manager shall initiate the implementation of the plan to satisfy the objectives and criteria set, exercising control over the process.

7.1.3.2 The manager shall monitor the execution of the process, providing both internal reporting of the process progress and external reporting to the acquirer as defined in the contract.

7.1.3.3 The manager shall investigate, analyze, and resolve the problems discovered during the execution of the process. The resolution of problems may result in changes to plans. It is the manager's responsibility to ensure the impact of any changes is determined, controlled, and monitored. Problems and their resolution shall be documented.

7.1.3.4 The manager shall report, at agreed points, the progress of the process, declaring adherence to the plans and resolving instances of the lack of progress. These include internal and external reporting as required by the organizational procedures and the contract.

Expected Output: ***Progress Monitoring Report***

7.1.4 Review and Evaluation

Required Input: ***Software Products***
Plans

7.1.4.1 It shall be ensured that the software products and plans are evaluated for satisfaction of requirements.

7.1.4.2 The manager shall assess the evaluation results of the software products, activities and tasks completed during the execution of the process for achievement of the objectives and completion of the plans.

Expected Output: ***Review records***

7.1.5 Closure

Required Input: ***Closure Request***

7.1.5.1. When all activities are completed, the management agency shall determine whether the process is complete as per contract or as per the organization's procedure.

7.1.5.2. The manager shall check the results and records of software products, activities, and tasks employed for completeness. These results and records shall be archived.

Expected Output: ***Archival of records***
Project Closure report

7.2 Infrastructure Process

This process addresses the need to establish and maintain the infrastructure needed for the software project. The infrastructure may include hardware, software, tools, process documents, and facilities for development, operation, or maintenance. This may be addressed in the Software Management Plan (SMP).

This process consists of the following activities:

- **Infrastructure Planning**
- **Establishment of the Infrastructure**
- **Maintenance of the Infrastructure**

7.2.1 Infrastructure Planning

Required Input: ***User Requirements***

7.2.1.1 The infrastructure should be defined and documented to meet the requirements of the processes, considering the applicable procedures, process documents, tools and techniques.

7.2.1.2 The establishment of the infrastructure shall be planned and documented.

Expected Output: ***Infrastructure Plan***

7.2.2 Establishment of Infrastructure

Required Input: ***Infrastructure Plan***

7.2.2.1 The configuration of the infrastructure shall be identified and documented. Functionality, performance, safety, security, availability, costs and time constraints should be considered.

7.2.2.2 It shall be ensured that the infrastructure be installed in time for execution of the relevant process.

Expected Output: ***Infrastructure Installation***

7.2.3 Maintenance of Infrastructure

Required Input: ***Infrastructure***

7.2.3.1 The infrastructure shall be maintained, monitored and modified as necessary to ensure that it continues to satisfy the process requirement. As part of maintaining the infrastructure, the extent to which the infrastructure is under configuration management shall be defined.

Expected Output: ***Maintenance Logs***

7.3 Improvement Process

The Improvement Process is a process for assessing, measuring, controlling, and improving a software life cycle process. The following activities are addressed in this process:

- **Process establishment**
- **Process assessment**
- **Process Improvement**

7.3.1 Process establishment.

7.3.1.1 The organization shall establish a suite of organizational processes for all software life cycle processes as they apply to its business activities. The processes and their application to specific cases shall be documented in organization's publications. As appropriate, a process control mechanism should be established to develop, monitor, control, and improve the process(es).

Expected Output: ***Software Process Document***

7.3.2 Process assessment

Required Input: ***Software Project Management Plan (SMP)***

7.3.2.1 A process assessment procedure should be developed, documented, and applied. Assessment records should be kept and maintained.

7.3.2.2 The measurements shall be reviewed periodically to ensure its continuing suitability and effectiveness based on assessment results.

Expected Output: ***Measurement records***

7.3.3 Process Improvement

This process addresses the improvements to each of its processes as a result of process assessment and review.

Required Input: ***Measurement records***

7.3.3.1 The organization shall effect such improvements to its processes as it determines to be necessary as a result of process assessment and review. Process documentation should be updated to reflect improvement in the organizational processes.

7.3.3.2 Historical, technical and evaluation data should be collected and analyzed to gain an understanding of the strengths and weaknesses of the employed processes. The results of this analysis shall be used as feedback to improve these processes, to recommend changes in the direction of the projects and to determine technology advancement needs.

7.3.3.3 Quality cost data should be collected, maintained, and used to improve the organization's processes as a management activity. These data shall serve the purpose of establishing the cost of both the prevention and resolution of problems and non-conformity in software products and services.

Expected Output: ***Analysis Report***

7.4 Training Process

This process addresses the aspects related to providing and maintaining trained personnel for carrying out the activities addressed in various processes. A review of the training requirements shall be carried out periodically to assess the types and levels of training and categories of personnel needing training.

The following activities are addressed in this process:

- **Planning of training activities**
- **Training material development**
- **Training plan implementation**

7.4.1 Planning of training activities

Required Input: ***Training Requirements***

7.4.1.1 A review of the project requirements shall be conducted to establish and make timely provision for acquiring or developing the resources and skills required by the management and technical staff. The types and levels of training and categories of personnel needing training shall be determined. A training plan, addressing implementation schedules, resource requirements, and training needs, should be developed and documented.

Expected Output: ***Training plan***

7.4.2 Training material development

Required Input: ***Training topics***

7.4.2.1 Training materials including presentation materials used in providing training has to be developed.

Expected Output: ***Training materials***

7.4.3 Training plan implementation

Required Input: ***Training Plan***

7.4.3.1 Training shall be carried out as prescribed by the training plan. Training records shall be maintained.

7.4.3.2 It should be ensured that the right mix and categories of appropriately trained personnel are available for the planned activities and tasks in a timely manner.

Expected Output: ***Training records***

8

Life Cycle for different categories of software in ISRO

The software developed in different ISRO centers are categorised into following seven categories.

- **Onboard software**
- **Checkout and Simulation software**
- **Launch Support and Test Facilities software**
- **Mission software**
- **Image Processing software**
- **Information Services software**
- **Scientific software**

The life cycle phases of various categories of software are represented in terms of a sequence chart and the corresponding linkage table. The sequence chart gives the sequence of activities to be carried out. The table provides activities and related inputs, outputs, responsibility, reviews and their linkage to IEEE/EIA 12207 clauses.

For implementation of the standard, the sequence chart shall be used in conjunction with the linkage table.

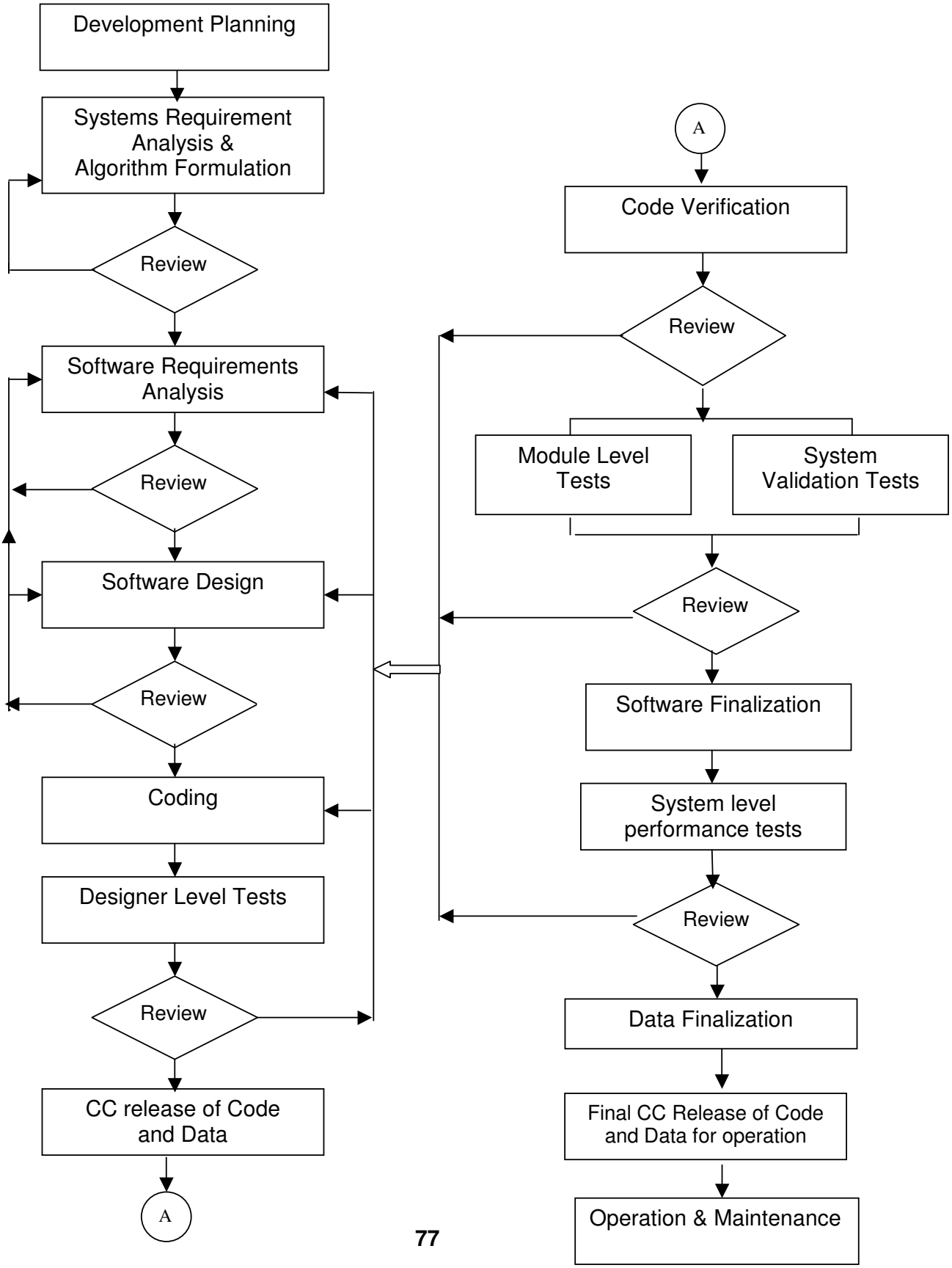
The Review can be either Formal Review (review conducted by a Central level identified committee/Team) or Internal Review (review conducted internally within a Division/Group/Area/Entity).

The supporting life cycle processes (section 6) are applicable to the activities as mentioned in this document.

THIS PAGE IS INTENTIONALLY LEFT BLANK

8.1 On-board software

8.1.1 Sequence of Activities



8.1.2 Linkage of activities in ISPD to IEEE 12207 (On-board software)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE 12207
Development Process						
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Formal Review	Development Planning (5.3.1)
2.	Systems Requirements	System Requirement Analysis /Algorithm Formulation	System Requirements Document (SRD) Algorithm Formulation/ Functional Requirement Document (FRD)	Developer	Formal Review	System Requirements Analysis (5.3.2) System Architectural design (5.3.3) Joint Review (6.6)
			System Test Plan	Developer	Internal review	
3.	SRD + FRD	Software Requirements Analysis	Software Requirements Specification (SRS)	Developer	Formal Review	Software Requirem-ents Analysis (5.3.4) Joint Review (6.6)
4.	SRS	Software Design	Software Design Document (SDD) User manual (If applicable)	Developer	Formal Review	Software Architectural Design (5.3.5) Software Detailed Design (5.3.6) Joint Review (6.6)
			Designer level Test Plan	Developer	Internal review	
5.	SRD/FRD+SRS + SDD + Designer Level Test Plan	Coding & Testing	Source Code & data. Designer level Test results	Developer	Internal Review	Software Coding and Testing (5.3.7)

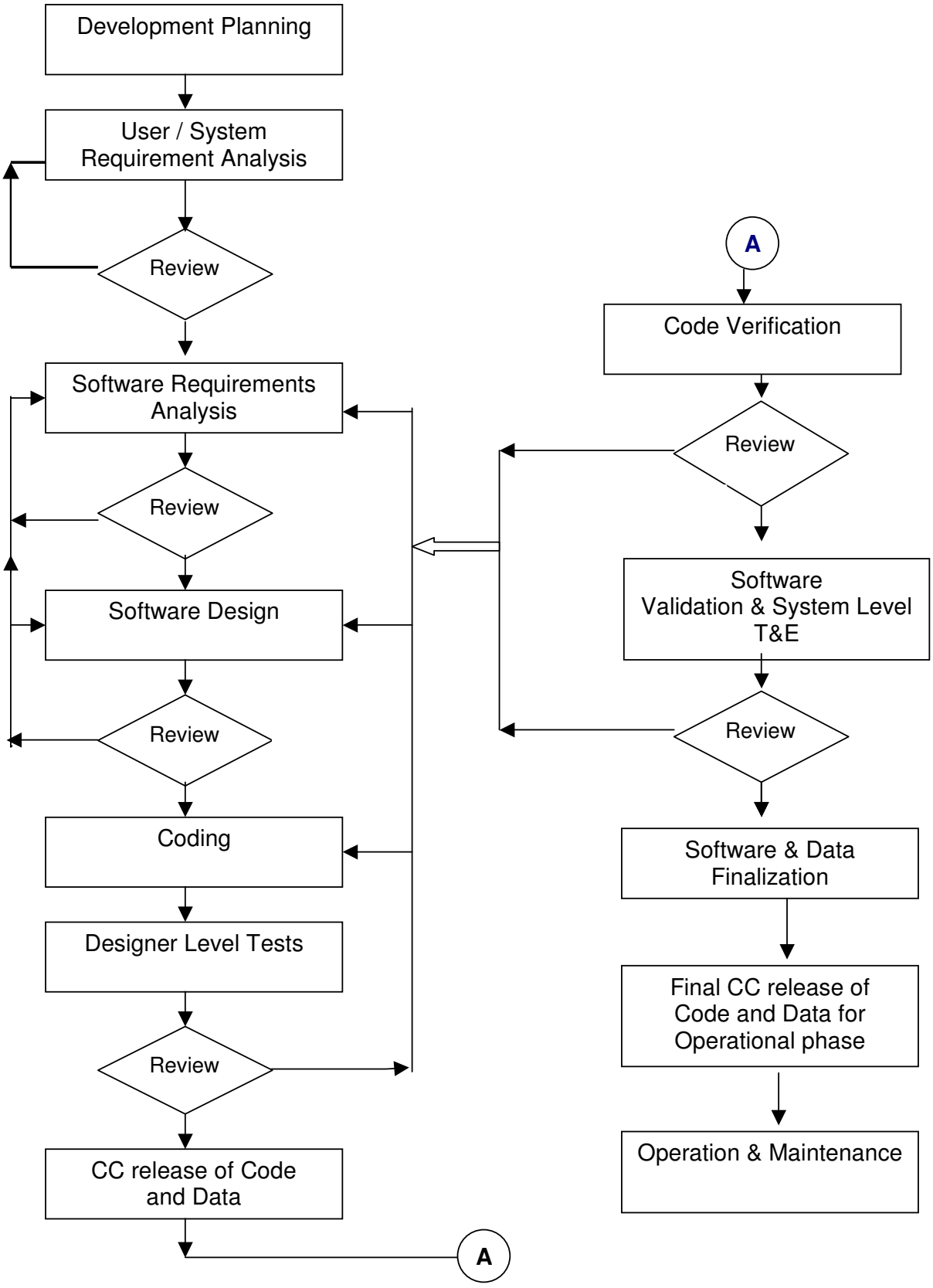
Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE 12207
6.	SRD/FRD+SRS+SDD+ Code & Data + Designer Level Test Plan.	Component level Integration & Testing (Designer level Test)	Component level Test results.	Developer	Internal Review	Software Integration (5.3.8) Software Qualification Testing (5.3.9)
7.	SRD/FRD+SRS+SDD + Code & Data + Integration Plan	Software Integration (H/W package level)	Integrated System Integration Test results	Integration Engineer	Formal Review	System Integration (5.3.10) System Qualification Testing (5.3.11) Joint Review (6.6)
8.	Version release note + Code + Data	Baseline & CC release of Code and Data for IV&V	Configuration Controlled code and data	CM	Configuration Audit	Configuration Management (6.2)
9.	FRD+SRS+SDD+ Configuration Controlled code & data	Code Verification	Verification Report	QA	Formal Review	Software Qualification Testing (5.3.9) -> Verification (6.4.2) Joint Review (6.6)
10.	FRD+SRS+SDD+ Module test plan+ Configuration Controlled code & data	Module Level Tests	Module Test report	QA	Formal Review	Software Qualification Testing (5.3.9) -> Validation (6.5) Joint Review (6.6)
11.	SRD+ Validation Test Plan + Integrated System	System Validation Tests	Validation Test report	QA, Simulation Agency, Developer	Formal Review	System Qualification Testing (5.3.11) ->Validation (6.5) Joint Review (6.6) Software Installation(5.3.12)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE 12207
12.	Performance Test cases + Integrated System	System level performance tests	Performance Test results	QA, Simulation Agency, Developer	Formal Review	
13.	Configuration Controlled code and data	Release of code and data	Released Software	Developer, CM	Internal Review	Software Acceptance Support (5.3.13)
Operation Process						
1.	Software Management Plan (SMP) + User manual	Operation Planning	Operation Procedure	User	Internal review	Process Implementation (5.4.1)
2.	Operational Test Plan	Operational Testing	Operational Test results / Validation Results	QA User	Internal Review	Operational Testing (5.4.2)
3.	Operation procedure/User Manual	Operation	Operation Results	User	Internal Review	System Operation (5.4.3)
4.	User Requests	Operation Support	User support log	User Developer	Internal Review	User support (5.4.4)
Maintenance Process						
1.	Software Management Plan (SMP)	Maintenance Planning	Software Maintenance Plan (SMP)	Maintainer	Internal Review	Process Implementation (5.5.1)
2.	Change Requirement / Problem Report	Change Review	Review Report	QA Maintainer	Formal Review	Problem & Modification Analysis (5.5.2), Joint Review(6.6)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE 12207
3.	Software Changes / Problem Solution	Change Implementation	Modified software	Maintainer QA	Internal Review	Modification Implementation (5.5.3) Maintenance Review / Acceptance (5.5.4) Development(5.3) Configuration Management (6.2)
4.	Migration Plan + Migration requirements	Migration	Migrated software	User Maintainer	Internal Review	Migration (5.5.5)
5.	Retirement request	Retirement	Retirement Plan, archives	User Maintainer	Internal Review	Software Retirement(5.5.6)

8.1 Checkout & Simulation Software

8.2.1 Sequence of activities



8.2.2 Linkage of activities in ISPD to IEEE 12207 (Checkout and Simulation Software)

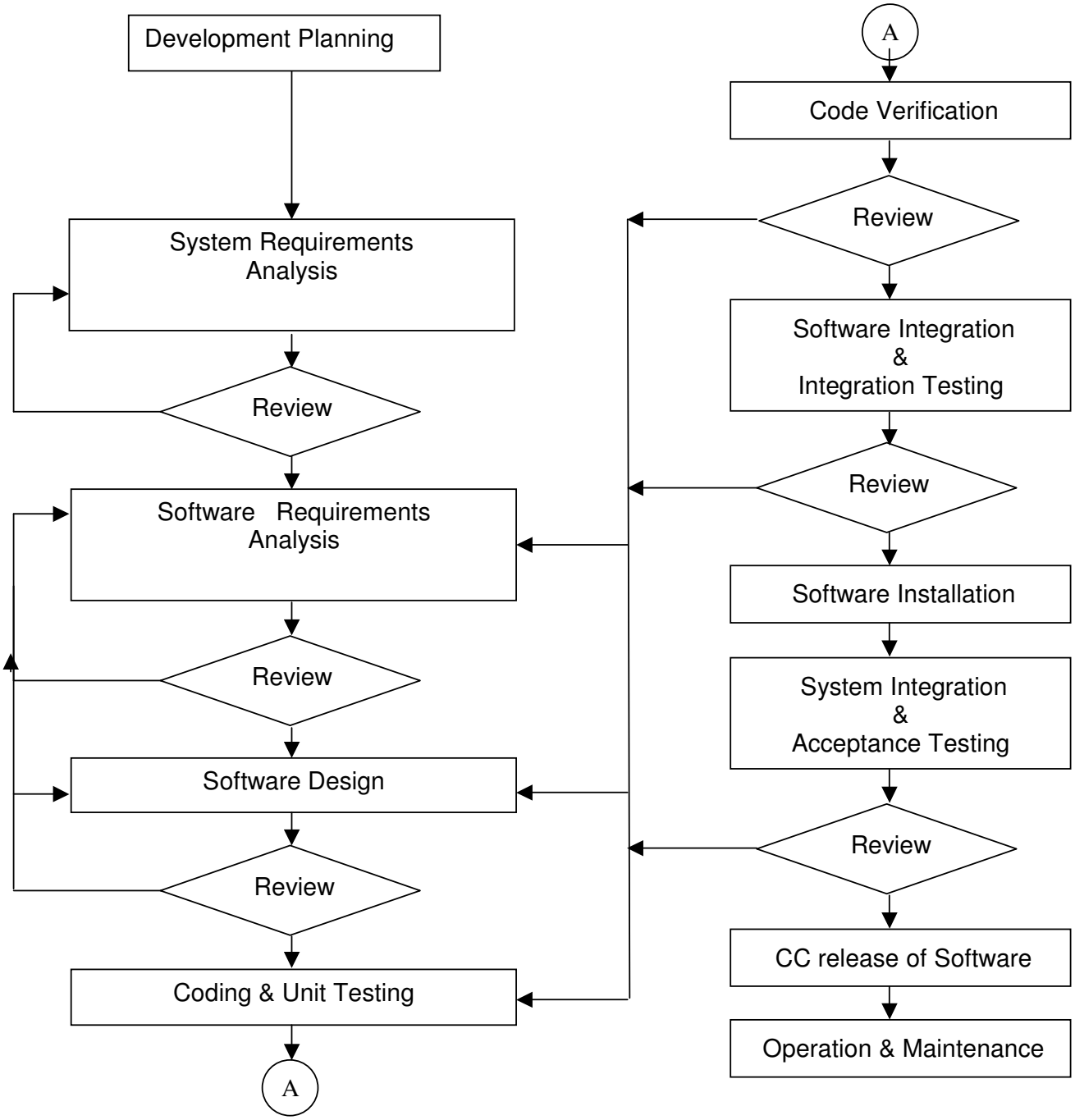
Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE 12207
Development Process						
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2.	Systems Requirements/User requirements	System Requirement Analysis	System Requirements Document (SRD) Algorithm Formulation/Functional Requirement Document (FRD) (if applicable)	Developer	Formal Review	System Requirements Analysis (5.3.2) System Architectural design (5.3.3) Joint Review(6.6)
			System Test Plan	Developer	Internal review	
3.	SRD/ User Requirement	Software Requirements Analysis	Software Requirements Specification (SRS)	Developer	Formal Review	Software Requirements Analysis (5.3.4) Joint Review(6.6)
4.	SRS	Software Design	Software Design Document (SDD) User manual (If applicable)	Developer	Formal Review	Software Architectural Design (5.3.5) Software Detailed Design (5.3.6) Joint Review(6.6)
			Designer level Test Plan	Developer	Internal review	
5.	SRD +SRS + SDD + Designer Level Test Plan	Coding & Testing	Source Code & data. Designer level Test results	Developer	Internal Review	Software Coding and Testing (5.3.7)

6.	SRD+SRS+SDD+ Code & Data + Designer Level Test Plan.	Component level Integration & Testing (Designer level Test)	Component level Test results.	Developer	Internal Review	Software Integration (5.3.8) Software Qualification Testing (5.3.9)
7.	SRD +SRS+SDD + Code & Data + Integration Plan	Software Integration (H/W package level)	Integrated System Integration Test results	Integration Engineer	Formal Review	System Integration (5.3.10) System Qualification Testing (5.3.11) Joint Review(6.6)
8.	Version release note + Code + Data	Baseline & CC release of Code and Data	Configuration Controlled code and data	CM Developer	Configuration Audit	Configuration Management (6.2)
9.	SRS+SDD+ Configuration Controlled code & data	Code Verification	Verification Report	QA	Formal Review	Verification (6.4.2) Joint Review(6.6)
10.	SRD+ Validation Test Plan + Integrated System	Software Validation Tests/System level T&E	Validation Test report/T&E Test results	QA, Simulation Agency, Developer	Formal Review	Software Qualification Testing (5.3.9 System Qualification Testing (5.3.11) - Validation (6.5) Joint Review(6.6)
11.	Version Release Note+Code+Data	Release of Final code & Data	Released Software	CM Developer	Configuration Audit	Configuration Management (6.2)
Operation Process						
1.	Software Management Plan (SMP) + User manual	Operation Planning	Operation Procedure/Operation Test plan	User , Developer	Internal Review	Process Implementation (5.4.1)

2.	Operational Test Plan	Operational Testing	Operation test results	QA User Developer	Internal Review	Operational Testing (5.4.2)
3.	Operation procedure/User Manual	Operation	Operation results	User Developer	Internal Review	System Operation (5.4.3)
4.	User Requests	Operation Support	User support log	User Developer	Internal Review	User support (5.4.4)
Maintenance Process						
1.	Software Management Plan (SMP)	Maintenance Planning	Software Maintenance Plan (SMP)	Maintainer	Internal Review	Process Implementation (5.5.1)
2.	Change Requirement / Problem Report	Change Review	Review Report	QA, Maintainer	Internal Review	Problem & Modification Analysis (5.5.2)
3.	Software Changes / Problem Solution	Change Implementation	Modified software	QA Maintainer	Internal Review	Modification Implementation (5.5.3) Maintenance Review / Acceptance (5.5.4) Development(5.3) Configuration Management(6.2)
4.	Migration Plan + Migration requirements	Migration	Migrated software	User Maintainer	Internal Review	Migration (5.5.5)
5.	Retirement request	Retirement	Retirement Plan, archives	User Maintainer	Internal Review	Software Retirement(5.5.6)

8.2 Launch Support and Test Facility Software

8.3.1 Sequence of activities



8.3.2 Linkage of activities in ISPD to IEEE 12207(Launch Support and Test Facility Software)

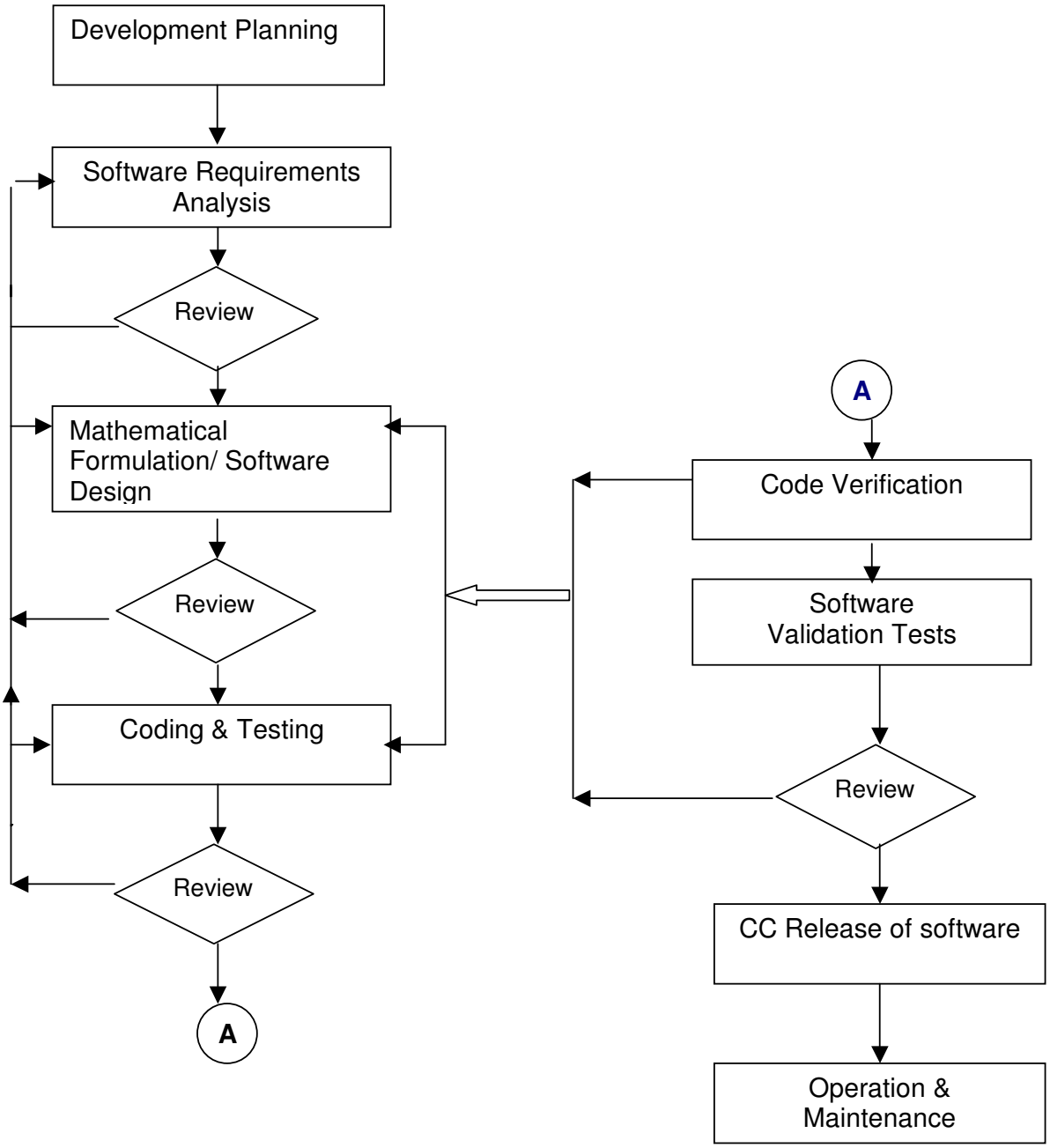
Sl.No.	Input	Activity	Output	Responsibility	Review	Linkage to IEEE STD 12207
1	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2	System Requirements + Mathematical Formulation(s)	System Requirements Analysis	System Requirement Specification , System Acceptance Test Plan	User	Formal	System Requirements Analysis (5.3.2) Joint Review (6.6)
3	System Requirement Specification	System Architecture Design	System Architecture Design Document	User	Formal	System Architectural Design (5.3.3) Joint Review (6.6)
4	System Requirement Specification + System Architecture Design Document	Software Requirement Analysis	Software Requirements Specification(SRS) Software Test Plan	Developer ,QA	Formal	System Requirements Analysis (5.3.4) Joint Review (6.6)
5	SRS	Software Architecture Design	Software Architecture Document , Integration Test Plan , User Manual	Developer , QA	Formal	Software Architectural Design (5.3.5) Joint Review (6.6)
6	SRS + Software Architecture Document +User Manual	Software Detailed Design	Software Design Document (SDD) , User Manual	Developer	Formal	Software Detailed Design (5.3.6) Joint Review (6.6)
1.	SRS + SDD + User Manual	Coding & Unit Testing	Source Code , User Manual, Unit Test results	Developer	Internal	Software Coding and Testing (5.3.7)
2.	SRS + SDD + Code & Data	Verification	Verification Report	QA / Peer Team	Formal	Verification (6.4) Joint Review (6.6)
3.	SRS + Software	Software Integration	Executables , User	Developer	Formal	Software Integration

Sl.No.	Input	Activity	Output	Responsibility	Review	Linkage to IEEE STD 12207
	Architecture Document + SDD + User Manual		Manual			(5.3.8) Joint Review (6.6)
4.	SRS + Integration Test Plan	Integration Testing	Integration Test Results	QA / Peer Team	Formal	Software Qualification Testing (5.3.9) Validation (6.5) Joint Review (6.6)
5.	Executables	Software Installation	User Operations Manual , Final UD	Developer	Formal	Software Installation (5.3.12) Joint Review (6.6)
6.	User Environment	System Integration	System Integration Report	Developer	Formal	System Integration (5.3.10) System Qualification Testing (5.3.11) Joint Review (6.6)
7.	System Acceptance Test Plan	Acceptance Testing + Training	Acceptance Report , User / Operator Training	User	Formal	Software Acceptance support (5.3.13) Joint Review (6.6)
8.	Version release note + Code + Data	Baseline & CC release of software	Configuration Controlled software	S/w CCB / QA	Configuration Audit	Configuration Management (6.2)
Operation Process						
1.	Software Management Plan (SMP) + User manual	Operation Planning	Operation Procedure/Operation Test plan	User Developer	Internal review	Process Implementation (5.4.1)
1.	Operation Test Plan+CC released software	Operational Testing	User Acceptance Report ,S/w Product release for operational use	User	Internal	Operational Testing (5.4.2)
2.	Operation	System Operation	Operate the s/w	User	Internal	System Operation

Sl.No.	Input	Activity	Output	Responsibility	Review	Linkage to IEEE STD 12207
	Procedure+User Manual +S/w Product released for operational use		,Problem reports and Modification requests to be communicated to Maintainer			(5.4.3) User Support (5.4.4)
Maintenance Process						
1.	Software Management Plan (SMP)	Maintenance Planning	Software Maintenance Plan (SMP)	Maintainer	Internal Review	Process Implementation (5.5.1)
1.	Software Maintenance Plan + Problem reports + Modification requests	Problem and Modification Analysis	Impact Analysis , Implementation options Documentation of Problem / Modification requests	Maintainer	Formal	Problem and Modification Analysis (5.5.2) Joint Review (6.6)
2.	Impact analysis report + Problem / Modification requests + Approval for selected modification option	Modification Implementation	T & E criteria Implementation of modified requirements	Maintainer & Developer	Formal	Modification Implementation (5.5.3) Joint Review (6.6) Development (5.3) Configuration Management (6.2)
3.	Modified Implementation (new version)	Maintenance Review / Acceptance	Approval for modification	Maintainer	Formal	Maintenance Review (5.5.4) Joint Review (6.6)
4.	Change in operational environment	Migration	Notification of migration plans and activities, Archiving of old environment	Maintainer	Formal	Migration (5.5.5) Joint Review (6.6)
5.	Request for retirement	S/w retirement	Notification of retirement plans, user training on new system + Archival of retired s/w	User	Formal	S/w Retirement (5.5.6) Joint Review (6.6)

8.3 Mission Software

8.4.1 Sequence of Activities



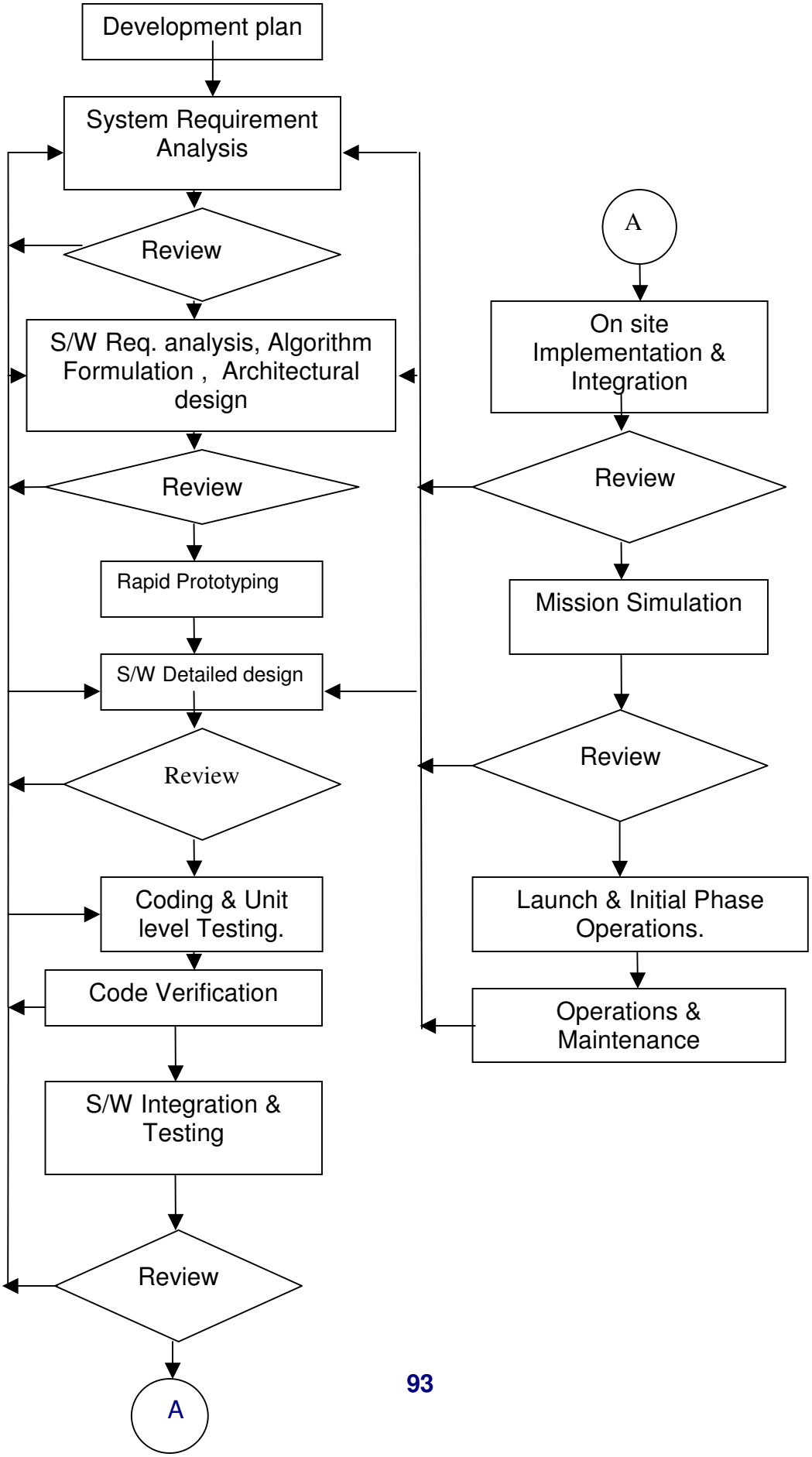
8.4.2 Linkage of activities in ISPD to IEEE 12207 (Mission Software)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to ISPD
Development Process						
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2.	User Requirements	Software Requirements Analysis	Software Requirements Specification (SRS)	Developer	Formal Review	Software Requirements Analysis (5.3.4) Joint Review(6.6)
3.	SRS	Mathematical Formulation / Software Design	Software Design Document (SDD) User manual (If applicable) Designer level Test plan	Developer	Formal Review	Software Architectural Design (5.3.5) Software Detailed Design (5.3.6) Joint Review(6.6)
4.	SRS+SDD + Designer Level Test Plan	Coding & Testing	Source Code & data. Designer level Test results	Developer	Internal Review	Software Coding and Testing (5.3.7) Software Integration (5.3.8)
5.	SRS+SDD+ Code & Data	Code Verification	Verification Report	Peer Team	Internal Review	Software Qualification Testing (5.3.9) -> Verification (6.4.2)
6.	SRS+SDD+ Validation Test Plan + Code & Data	Software Validation Tests	Validation Test report	Peer Team	Internal Review	Software Qualification Testing (5.3.9) -> Validation (6.5)

7.	Version release note + Code + Data	Baseline & CC release of Software	Configuration Controlled software User manual	CM	Configuration Audit	Configuration Management (6.2) Software Acceptance Support (5.3.13)
Operation Process						
1.	Software Management Plan (SMP) + User manual	Operation Planning	Operation Procedure	User	Internal review	Process Implementation (5.4.1)
2.	T&E Plan / validation Test Plan	Operational Testing	Test results / Validation Results	QA	Internal Review	Operational Testing (5.4.2)
3.	Operation procedure Test Cases	Mission Analysis	Analysis Results	User	Internal Review	System Operation (5.4.3) User support (5.4.4)
Maintenance Process						
1.	Software Management Plan (SMP)	Maintenance Planning	Software Maintenance Plan (SMP)	Maintainer	Internal Review	Process Implementation (5.5.1)
2.	Change Requirement / Problem Report	Change Review	Review Report	QA	Internal Review	Problem & Modifica-tion Analysis (5.5.2)
3.	Software Changes / Problem Solution	Change Implementation	Modified software	Maintainer QA	Internal Review	Modification Implementation (5.5.3) Maintenance Review / Acceptance (5.5.4) Development(5.3) Configuration Management(6.2)
4.	Migration Plan + Migration requirements	Migration	Migrated software	Development Team	Internal Review	Migration (5.5.5)
5.	Retirement request	Retirement	Retirement Plan, archives	User Maintainer	Internal Review	Software Retirement(5.5.6)

8.5 Image Processing Software

8.5.1 Sequence of activities



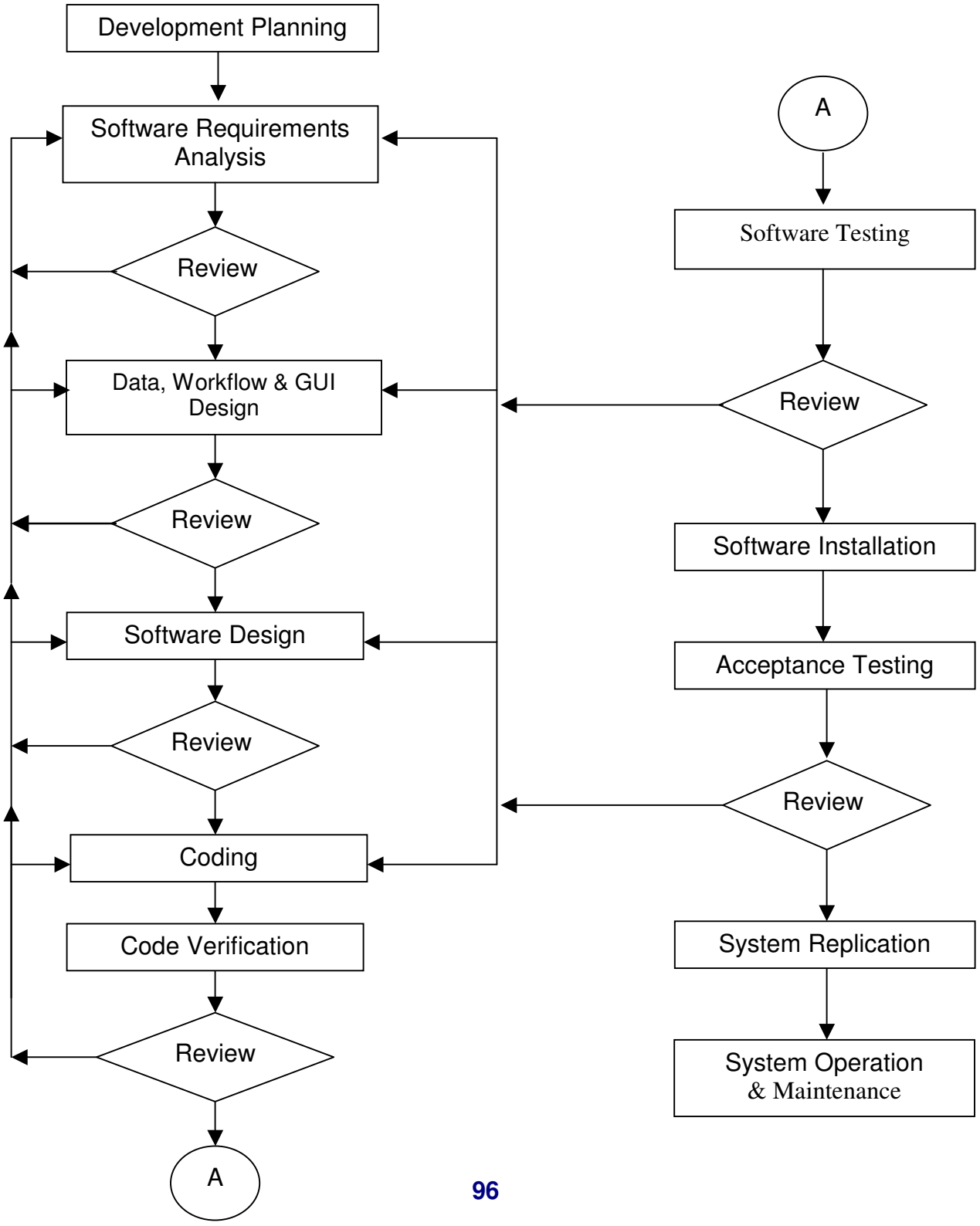
8.5.2 Linkages of activities in ISPD to IEEE 12207 (Image Processing Software)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to ISPD
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2	Systems Requirements	System Requirement Analysis	Software Requirement Document	User, Developer	Formal Review	System Requirements Analysis (5.3.2) Joint Review(6.6)
3	Software Requirement Document	S/W requirements Analysis, Algorithm Formulation, S/W Architectural Design & I/F Identification.	Architectural Design Document	User, Developer	Formal Review	Software Requirements Analysis (5.3.4) Software Architectural Design (5.3.5) Joint Review(6.6)
			System Test Plan	Developer	Internal review	
4	Architectural Design Document	Software Detailed Design Interface identification and finalization	Detailed Design document(SDD) Designer level Test Plan ICD	Developer	Formal Review	Software Detailed Design (5.3.6) Joint Review(6.6)
5	Detailed Design document+ ICD	Coding & Unit testing	Source Code & data. Unit level Test results	Developer	Internal review	Software Coding and Testing (5.3.7)
6.	SDD + ICD + Unit Code +	Code Verification and Audit	Verification Report	Peer Team	Internal Review	Software Qualification Testing (5.3.9) -> Verification (6.4.2)

7.	SDD + ICD + Code & Data	Sub-system level Integration & Testing (Designer level Test)	Test results	Developer & Internal T&E committee	Internal Review	Software Integration (5.3.8) Software Qualification Testing (5.3.9)
8.	SDD + ICD + sub-system level Code & Data + Implementation Plan+ Test Plans	Deployment, System Testing and Verification	Integrated software, User Manual, Test Plan, Designer Test Results, Implementation Report	Developer	Developer	System Integration (5.3.10)
9.	Systems Requirements + SRS + S/W Package + Data + Test plan & procedures Document + Designer Test Results + Implementation Report	Test And Evaluation	Test Report Certified System for Operations	T & E committee	Formal Review	System Qualification Testing (5.3.11) Joint Review (6.6)
9.	Simulation plan	Mission simulation	Simulation report	Developer, User	Formal review	System Qualification Testing (5.3.11) Joint Review (6.6)
10	User Manual	Operations	Acceptance Report	User	Formal Review	Software Acceptance support (5.3.13) Joint Review (6.6)
11	Operational Anomaly	Software maintenance .	Modified Software	Maintainer	CMB	Configuration Management (6.2)

8.6 Information Services Software

8.6.1 Sequence of activities



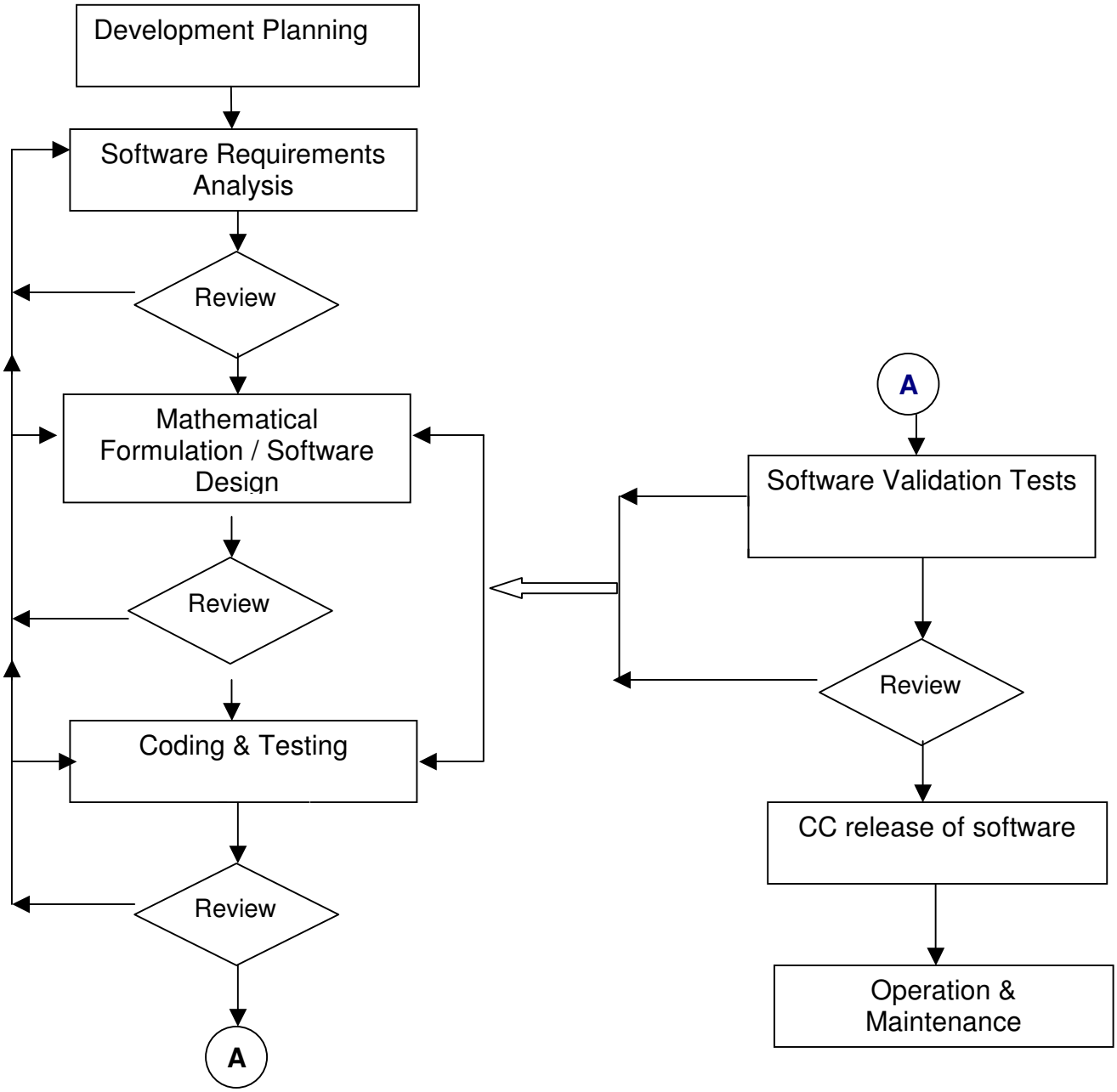
8.6.2 Linkage of activities in ISPD to IEEE 12207 (Information Services software)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to IEEE-STD 12207
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2.	System Requirements Document (SRD)	Software Requirements Analysis	Software Requirements Specification (SRS)	Developer	Formal Review	System Requirem-ents Analysis (5.3.2)
			System Test Plan	Developer	Internal Review	Software Requirements Analysis (5.3.4)
3	SRS	Data, Workflow & GUI Design	Database Schema Software Architectutre Design Document(SAD)	Developer	Formal Review	Software Architectural Design (5.3.5)
4	Database Schema SRS + SAD	Software Design	Software Design Document (SDD) Preliminary User Manual	Developer	Formal Review	Software Detailed Design (5.3.6) Joint Review(6.6)
5	Database Schema SAD + SDD	Coding	Source Code & Database	Developer	Internal Review	Software Coding & Testing (5.3.7)
6	SRS Source Code & Database	Software Testing	Test Results	QA	Formal Review	Software Qualification Testing (5.3.9) Joint Review(6.6)

7	Source Code & Database	Software Installation	Working System	Developer	Formal Review	Software Installation (5.3.12) Joint Review(6.6)
8	SRD Working System	Acceptance Testing	Test Report	User	Formal Review	Software Acceptance Support (5.3.13) Joint Review(6.6)
9	Working System	System Replication	Deployed System	Developer	Internal Review	Operation (5.4)
10	Deployed System	System Operation & Maintenance	Operational System	User	Internal Review	System Operation (5.4.3)

8.7 Scientific Software

8.7.2 Sequence of activities



8.7.2 Linkage of activities in ISPD to IEEE 12207 (Scientific Software)

Sl. No	Input	Activity	Output	Responsibility	Review	Linkage to ISPD
Development Process						
1.	Software Management Plan (SMP)	Development Planning	Software Development Plan (SDP)	Developer	Internal Review	Development Planning (5.3.1)
2.	User Requirements	Software Requirements Analysis	Software Requirements Specification (SRS)	Developer	Internal Review	Software Requirem-ents Analysis (5.3.4)
3.	SRS	Mathematical Formulation / Software Design	Software Design Document (SDD) User manual (If applicable) Designer level Test plan	Developer	Internal Review	Software Architectural Design (5.3.5) Software Detailed Design (5.3.6)
4.	SRS+SDD + Designer Level Test Plan	Coding & Testing	Source Code & data. Designer level Test results	Developer	Internal Review	Software Coding and Testing (5.3.7) Software Integration (5.3.8)
6.	SRS+SDD+ Validation Test Plan + Code & Data	Software Validation Tests	Validation Test report	Peer Team	Internal Review	Software Qualification Testing (5.3.11) -> Validation (6.5)
7.	Version release note + Code + Data	Baseline & CC release of Software	Configuration Controlled software User manual	CM	Configuration Audit	Configuration Management (6.2) Software Acceptance Support (5.3.13)

Operation Process						
1.	Software Management Plan (SPMP) / User Manual	Operation Planning	Operation Procedure	User	Internal review	Process Implementation (5.4.1)
2.	Validation Test Plan	Software validation Tests	Validation Results	User	Internal Review	Operational Testing (5.4.2)
3.	Analysis inputs	Analysis	Analysis Results	User	Internal review	System Operation (5.4.3) User support (5.4.4)
Maintenance Process						
1.	Software Management Plan (SMP)	Maintenance Planning	Software Maintenance Plan (SMP)	User	Internal Review	Process Implementation (5.5.1)
2.	Change Requirement / Problem Report	Change Review	Review Report	QA	Internal Review	Problem & Modifica-tion Analysis (5.5.2)
3.	Software Changes / Problem Solution	Change Implementation	Modified software	Developer / User QA	Internal Review	Modification Implementation (5.5.3) Maintenance Review / Acceptance (5.5.4) Development (5.3) Configuration Management(6.2)
4.	Migration Plan + Migration requirements	Migration	Migrated software	Development Team / User	Internal Review	Migration (5.5.5)
5.	Retirement request	Retirement	Retirement Plan, archives	User Maintainer	Internal Review	Software Retirement(5.5.6)

Table-1

Process Applicability Matrix of ISPD(Sec 5.1 and 5.2) to ISRO Software

***Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable**

Sl. No	ISPD Process	Related Activities		Related tasks	Applicability		
					Outsourced	Inter Centre	Intra Centre
1	**Acquisition**	5.1.1	Generation of indent proposal/Request for information-RFI	5.1.1.1 5.1.1.2 5.1.1.3 5.1.1.4 5.1.1.5 5.1.1.6 5.1.1.7 5.1.1.8 5.1.1.9	M M T T T M M M M	M M T T T M M M M	M M T T T M M M M
		5.1.2	Request for Proposal Preparation/Tender preparation	5.1.2.1 5.1.2.2 5.1.2.3 5.1.2.4	M M M M	T M M M	T M T M
		5.1.3	Contract Preparation & Update	5.1.3.1 5.1.3.2 5.1.3.3 5.1.3.4 5.1.3.5	M M T M M	T T T T T	T T T T T
		5.1.4	Supplier Monitoring	5.1.4.1 5.1.4.2	M M	M M	M M
		5.1.5	Acceptance and Completion	5.1.5.1 5.1.5.2 5.1.5.3	M M M	M M M	M M M

Process Applicability Matrix of ISPD(Sec 5.1 and 5.2) to ISRO Software

***Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable**

2	**Supply**	5.2.1	Initiation	5.2.1.1 5.2.1.2	M M	M M	M M
		5.2.2	Preparation of Response	5.2.2.1	M	T	T
		5.2.3	Contract	5.2.3.1 5.2.3.2	M M	T T	T T
		5.2.4	Planning	5.2.4.1 5.2.4.2 5.2.4.3 5.2.4.4 5.2.4.5	M M M M M	M M M M M	M M M M M
		5.2.5	Execution & Control	5.2.5.1 5.2.5.2 5.2.5.3 5.2.5.4 5.2.5.5 5.2.5.6	M M M M M M	M M T M M M	M M T M M M
		5.2.6	Review & Evaluation	5.2.6.1 5.2.6.2 5.2.6.3 5.2.6.4 5.2.6.5 5.2.6.6	M M M M M M	T M M M M T	T M T T M T
		5.2.7	Delivery & Completion	5.2.7.1 5.2.7.2	M M	M M	M M

Table-2

Process Applicability matrix of ISPD(Sec 5.3 to 7.4) to different categories of software

***Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable**

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
3	Development	5.3.1	Development Planning	5.3.1.1	M	M	M	M	M	M	M
				5.3.1.2	M	M	M	M	M	M	M
				5.3.1.3	M	M	M	M	M	M	M
				5.3.1.4	M	M	M	M	M	M	M
				5.3.1.5	M	M	M	M	M	T	M
		5.3.2	System Requirements Analysis	5.3.2.1	M	M	T	T	T	T	NA
				5.3.2.2	M	M	T	T	T	T	NA
		5.3.3	System Architectural design	5.3.3.1	M	M	T	T	T	T	NA
				5.3.3.2	M	M	T	T	T	T	NA
		5.3.4	Software Requirements Analysis	5.3.4.1	M	M	M	M	M	M	M
				5.3.4.2	M	M	M	M	M	M	M
				5.3.4.3	M	M	M	M	M	M	M
		5.3.5	Software Architectural Design	5.3.5.1	M	M	M	M	M	M	M
				5.3.5.2	M	M	M	M	M	M	M
				5.3.5.3	M	M	M	M	M	M	T
				5.3.5.4	T	T	T	T	T	T	T
				5.3.5.5	T	M	M	T	T	T	T
				5.3.5.6	M	M	M	M	M	M	M
				5.3.5.7	M	M	M	M	M	T	M
		5.3.6	Software Detailed Design	5.3.6.1	M	M	M	M	M	M	M
				5.3.6.2	M	M	M	M	M	M	M
				5.3.6.3	M	M	M	M	M	M	M
				5.3.6.4	T	T	T	T	T	T	T
				5.3.6.5	M	T	T	T	T	T	T
				5.3.6.6	M	T	T	T	T	T	T
				5.3.6.7	M	M	M	M	M	M	M
				5.3.6.8	M	M	M	M	M	M	M

*Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
		5.3.7	Software Coding and Testing	5.3.7.1	M	M	M	M	M	M	M
				5.3.7.2	M	M	M	M	M	M	M
				5.3.7.3	T	T	T	T	T	T	T
				5.3.7.4	M	T	T	T	T	T	T
				5.3.7.5	M	M	M	M	M	M	M
		5.3.8	Software Integration	5.3.8.1	M	M	M	M	M	T	T
				5.3.8.2	M	M	M	M	M	T	M
				5.3.8.3	M	M	M	M	T	T	M
				5.3.8.4	M	M	M	M	M	T	M
				5.3.8.5	M	M	M	M	M	T	M
				5.3.8.6	M	M	M	M	M	T	M
		5.3.9	Software Qualification Testing	5.3.9.1	M	M	M	M	M	M	M
				5.3.9.2	M	M	M	M	M	T	M
				5.3.9.3	M	M	M	M	M	M	M
				5.3.9.4	T	T	T	T	T	T	T
				5.3.9.5	T	T	T	T	T	T	T
		5.3.10	System Integration	5.3.10.1	M	M	M	M	M	T	NA
				5.3.10.2	M	M	M	M	M	T	NA
				5.3.10.3	M	M	M	M	M	T	NA
		5.3.11	System Qualification Testing	5.3.11.1	M	M	M	M	M	T	NA
				5.3.11.2	M	M	M	M	M	T	NA
				5.3.11.3	T	T	T	T	T	T	NA
				5.3.11.4	T	T	T	T	T	T	NA
		5.3.12	Software Installation	5.3.12.1	M	T	T	M	M	M	T
				5.3.12.2	M	T	T	M	M	M	T

*Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
		5.3.13	Software Acceptance Support	5.3.13.1	M	M	M	M	M	M	M
				5.3.13.2	M	M	M	M	M	M	M
				5.3.13.3	M	M	M	M	M	T	M
4	Operation	5.4.1	Operation Planning	5.4.1.1	M	M	T	M	M	M	T
				5.4.1.2	M	M	T	M	M	M	T
				5.4.1.3	M	M	T	M	M	T	T
		5.4.2	Operational testing	5.4.2.1	M	M	T	M	M	T	T
				5.4.2.2	M	M	T	M	M	T	T
		5.4.3	System operation	5.4.3.1	M	M	T	M	M	M	T
		5.4.4	User support	5.4.4.1	M	M	T	M	M	M	T
				5.4.4.2	M	M	T	M	M	M	T
				5.4.4.3	M	M	T	M	M	T	T
5	Maintenance	5.5.1	Maintenance Planning	5.5.1.1	M	M	M	M	M	M	T
				5.5.1.2	M	M	M	M	M	M	T
				5.5.1.3	M	M	M	M	M	M	T
		5.5.2	Problem and Modification Analysis	5.5.2.1	M	M	M	M	M	M	T
				5.5.2.2	M	M	M	M	M	M	T
				5.5.2.3	M	M	M	M	M	M	T
				5.5.2.4	M	M	M	M	M	M	T
				5.5.2.5	M	M	M	M	M	M	T
		5.5.3	Modification Implementation	5.5.3.1	M	M	M	M	M	M	T
				5.5.3.2	M	M	M	M	M	M	T
		5.5.4	Maintenance Review / acceptance	5.5.4.1	M	M	M	M	M	M	T
				5.5.4.2	M	M	M	M	M	M	T
		5.5.5	Migration	5.5.5.1	NA	T	T	T	T	T	T
				5.5.5.2	NA	T	T	T	T	T	T
				5.5.5.3	NA	T	T	T	T	T	T
				5.5.5.4	NA	T	T	T	T	T	T
				5.5.5.5	NA	T	T	T	T	T	T
				5.5.5.6	NA	T	T	T	T	T	T
				5.5.5.7	NA	T	T	T	T	T	T

*Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
		5.5.6	Software Retirement	5.5.6.1	T	T	T	T	T	T	T
				5.5.6.2	T	T	T	T	T	T	T
				5.5.6.3	T	T	T	T	T	T	T
				5.5.6.4	T	T	T	T	T	T	T
				5.5.6.5	T	T	T	T	T	T	T
6	Documentation	6.1.1	Documentation Planning	6.1.1.1	M	M	M	M	M	M	M
		6.1.2	Design and Development	6.1.2.1	M	M	M	M	M	M	M
				6.1.2.2	M	M	M	M	M	M	M
				6.1.2.3	M	M	M	M	M	M	M
		6.1.3	Production	6.1.3.1	M	M	M	M	M	M	M
				6.1.3.2	M	M	M	M	M	M	M
		6.1.4	Maintenance	6.1.4.1	M	M	M	M	M	M	M
7	Configuration Management	6.2.1	Configuration Management Planning	6.2.1.1	M	T	T	T	T	M	T
		6.2.2	Configuration Identification	6.2.2.1	M	T	T	T	T	M	T
		6.2.3	Configuration Control	6.2.3.1	M	T	T	T	T	M	T
		6.2.4	Configuration Status Accounting	6.2.4.1	M	T	T	T	T	M	T
		6.2.5	Configuration Evaluation	6.2.5.1	M	T	T	T	T	M	T
		6.2.6	Release Management & Delivery	6.2.6.1	M	T	T	T	T	M	T
8	Quality Assurance	6.3.1	Software Quality Assurance Planning	6.3.1.1	M	T	T	T	T	T	T
				6.3.1.2	M	T	T	T	T	T	T
				6.3.1.3	M	T	T	T	T	T	T
				6.3.1.4	M	T	T	T	T	T	T
				6.3.1.5	M	T	T	T	T	T	T
				6.3.1.6	M	T	T	T	T	T	T
		6.3.2	Product Assurance	6.3.2.1	M	T	T	T	T	M	T
				6.3.2.2	M	T	T	T	T	M	T
				6.3.2.3	M	T	T	T	T	M	T

*Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
		6.3.3	Process Assurance	6.3.3.1	M	T	T	T	T	M	T
				6.3.3.2	M	T	T	T	T	M	T
				6.3.3.3	M	T	T	T	T	T	T
				6.3.3.4	M	T	T	T	T	T	T
				6.3.3.5	M	T	T	T	T	T	T
				6.3.3.6	M	T	T	T	T	T	T
		6.3.4	Assurance of Quality Systems	6.3.4.1	M	T	T	T	T	T	T
9	Verification	6.4.1	Software Verification Planning	6.4.1.1	M	T	T	T	T	M	T
				6.4.1.2	M	T	T	T	T	T	T
				6.4.1.3	M	T	T	T	T	T	T
				6.4.1.4	M	T	T	T	T	T	T
				6.4.1.5	M	T	T	T	T	M	T
				6.4.1.6	M	T	T	T	T	M	T
		6.4.2	Verification	6.4.2.1	M	T	T	T	T	T	T
				6.4.2.2	M	T	T	T	T	M	T
				6.4.2.3	M	T	T	T	T	T	T
				6.4.2.4	M	T	T	T	T	M	T
				6.4.2.5	M	T	T	T	T	M	T
				6.4.2.6	M	T	T	T	T	M	T
				6.4.2.7	M	T	T	T	T	M	T
10	Validation	6.5.1	Validation Test Planning	6.5.1.1	M	T	T	T	T	M	T
				6.5.1.2	M	T	T	T	T	T	T
				6.5.1.3	M	T	T	T	T	T	T
				6.5.1.4	M	T	T	T	T	M	T
				6.5.1.5	M	T	T	T	T	M	T
		6.5.2	Validation	6.5.2.1	M	T	T	T	T	M	T
				6.5.2.2	M	T	T	T	T	M	T
				6.5.2.3	M	T	T	T	T	M	T
				6.5.2.4	M	T	T	T	T	M	T
				6.5.2.5	M	T	T	T	T	M	T

***Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable**

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
11	Joint Review	6.6.1	Review Planning	6.6.1.1	M	M	M	M	M	M	M
				6.6.1.2	M	M	M	M	M	M	M
				6.6.1.3	M	M	M	M	M	M	M
				6.6.1.4	M	M	M	M	M	M	M
				6.6.1.5	M	M	M	M	M	M	M
				6.6.1.6	M	M	M	M	M	M	M
		6.6.2	Project Management Reviews	6.6.2.1	M	M	M	M	M	M	M
		6.6.3	Technical Reviews	6.6.3.1	M	M	M	M	M	M	M
12	Audit	6.7.1	Audit Planning	6.71.1	T	T	T	T	T	T	T
				6.7.1.2	T	T	T	T	T	T	T
				6.7.1.3	T	T	T	T	T	T	T
				6.7.1.4	T	T	T	T	T	T	T
				6.7.1.5	T	T	T	T	T	T	T
				6.7.1.6	T	T	T	T	T	T	T
				6.7.1.7	T	T	T	T	T	T	T
		6.7.2	Audit	6.7.2.1	T	T	T	T	T	T	T
13	Problem Resolution	6.8.1	Problem Resolution Planning	6.8.1.1	M	M	M	M	M	M	M
		6.8.2	Problem Resolution	6.8.2.1	M	M	M	M	M	M	M
14	Management	7.1.1	Initiation and scope definition	7.1.1.1	T	T	T	T	T	T	T
				7.1.1.2	T	T	T	T	T	T	T
				7.1.1.3	T	T	T	T	T	T	T
		7.1.2	Management Planning	7.1.2.1	T	T	T	T	T	T	T
		7.1.3	Execution and Control	7.1.3.1	T	T	T	T	T	T	T
				7.1.3.2	T	T	T	T	T	T	T
				7.1.3.3	T	T	T	T	T	T	T
				7.1.3.4	T	T	T	T	T	T	T
		7.1.4	Review and Evaluation	7.1.4.1	T	T	T	T	T	T	T
				7.1.4.2	T	T	T	T	T	T	T
		7.1.5	Closure	7.1.5.1	T	T	T	T	T	T	T
				7.1.5.2	T	T	T	T	T	T	T

***Applicability : M-Mandatory, T-Tailorable, NA-Not Applicable**

Sl. No	ISPD Process	Related Activities		Related tasks	*Applicability to software category						
					Onboard	Checkout & Simulation	Launch & Test Facilities	Image Processing	Mission	Information Services	Scientific
15	Infrastructure	7.2.1	Infrastructure Planning	7.2.1.1 7.2.1.2	T T	T T	T T	M M	M M	T T	T T
		7.2.2	Establishment of the Infrastructure	7.2.2.1 7.2.2.2	T T	T T	T T	M M	M M	T T	T T
		7.2.3	Maintenance of the Infrastructure	7.2.3.1	T	T	T	M	M	T	T
16	Improvement	7.3.1	Process Establishment	7.3.1.1	T	T	T	T	T	T	T
		7.3.2	Process Assessment	7.3.2.1 7.3.2.2	T T	T T	T T	T T	T T	T T	T T
		7.3.3	Process Improvement	7.3.3.1 7.3.3.2 7.3.3.3	T T T	T T T	T T T	T T T	T T T	T T T	T T T
17	Training	7.4.1	Planning of Training activities	7.4.1.1	T	T	T	T	T	T	T
		7.4.2	Training Material development	7.4.2.1	T	T	T	T	T	T	T
		7.4.3	Training Plan Implementation	7.4.3.1 7.4.3.2	T T	T T	T T	T T	T T	T T	T T

ANNEXURE-A
Documentation Templates

Sl.No	Document	IEEE Reference
1.	Software Project Management Plan (SMP)	IEEE Std 1058.1-1987
2.	Software Quality Assurance Plan (SQAP)	IEEE Std 730™-2002
3.	Software Verification & Validation Plan (SVVP)	IEEE Std 1012-1998
4.	Software Configuration Management Plan(SCMP)	IEEE Std 828-1998
5.	System Requirements Document (SRD)	IEEE Std 1233, 1998
6.	Software Requirements Specification (SRS)	IEEE Std 830-1998
7.	Software Design Document (SDD)	IEEE Std 1016-1998
8.	Software Test Plan(STP)	IEEE Std 1008-1997(R2002)
9.	Software Maintenance Plan (SMNP)	IEEE Std 1219-1998
10.	Software Development Plan(SDP)	IEEE Std 1074-1997