

# Common Framework for Life Cycles and their Processes

*System and Software Life Cycle Processes*

## Contents

1	CHANGELOG . . . . .	3
2	INTRODUCTION . . . . .	4
3	UNDERSTANDING AND APPLYING THIS STANDARD . . . . .	5
3.1	RELATIONSHIP BETWEEN SYSTEMS AND SOFTWARE . . . . .	5
3.2	ORGANIZATION-LEVEL AND PROJECT-LEVEL ADOPTION . . . . .	5
3.3	TEMPORAL RELATIONSHIPS AMONG PROCESSES . . . . .	5
3.4	EVALUATION VS VERIFICATION AND VALIDATION . . . . .	6
3.5	CRITERIA FOR PROCESSES . . . . .	6
3.6	DESCRIPTION AND GENERAL CHARACTERISTICS OF PROCESSES . . . . .	6
3.7	DECOMPOSITION OF PROCESSES . . . . .	7
3.8	LIFE CYCLE MODELS AND STAGES . . . . .	7
4	STRUCTURE, ORGANIZATION, AND IMPLEMENTATION . . . . .	8
4.1	CATEGORIES OF LIFE CYCLE PROCESSES . . . . .	8
4.2	SUMMARY OF LIFE CYCLE PROCESSES . . . . .	8
4.2.1	AGREEMENT PROCESSES . . . . .	9
4.2.2	ORGANIZATIONAL PROJECT-ENABLING PROCESSES . . . . .	9
4.2.3	PROJECT PROCESSES . . . . .	9
4.2.4	ORGANIZATIONAL PROCESSES . . . . .	9
4.2.5	TECHNICAL PROCESSES . . . . .	9
4.2.6	SOFTWARE IMPLEMENTATION PROCESSES . . . . .	9
4.2.7	SOFTWARE SUPPORT PROCESSES . . . . .	9
4.2.8	SOFTWARE REUSE PROCESSES . . . . .	9
4.3	STANDARD AS PROCESS REFERENCE MODEL (PRM) . . . . .	9
5	SYSTEM CONTEXT PROCESSES . . . . .	11
5.1	AGREEMENT PROCESSES . . . . .	11
5.1.1	ACQUISITION PROCESS . . . . .	12
5.1.1.1	PURPOSE . . . . .	12
5.1.1.2	OUTCOMES . . . . .	12
5.1.1.3	ACTIVITIES AND TASKS . . . . .	12
5.1.2	SUPPLY PROCESS . . . . .	15
5.1.2.1	PURPOSE . . . . .	15
5.1.2.2	OUTCOMES . . . . .	15
5.1.2.3	ACTIVITIES AND TASKS . . . . .	15
5.2	ORGANIZATIONAL PROJECT-ENABLING PROCESSES . . . . .	18
5.2.1	LIFE CYCLE MODEL MANAGEMENT PROCESS . . . . .	19
5.2.1.1	PURPOSE . . . . .	19
5.2.1.2	OUTCOMES . . . . .	19
5.2.1.3	ACTIVITIES AND TASKS . . . . .	19
5.2.2	INFRASTRUCTURE MANAGEMENT PROCESS . . . . .	20
5.2.2.1	PURPOSE . . . . .	20
5.2.2.2	OUTCOMES . . . . .	20
5.2.2.3	ACTIVITIES AND TASKS . . . . .	20

5.2.3	PROJECT PORTFOLIO MANAGEMENT PROCESS . . . . .	21
5.2.3.1	PURPOSE . . . . .	21
5.2.3.2	OUTCOMES . . . . .	21
5.2.3.3	ACTIVITIES AND TASKS . . . . .	21
5.2.4	HUMAN RESOURCE MANAGEMENT PROCESS . . . . .	23
5.2.4.1	PURPOSE . . . . .	23
5.2.4.2	OUTCOMES . . . . .	23
5.2.4.3	ACTIVITIES AND TASKS . . . . .	23
5.2.5	QUALITY MANAGEMENT PROCESS . . . . .	25
5.2.5.1	PURPOSE . . . . .	25
5.2.5.2	OUTCOMES . . . . .	25
5.2.5.3	ACTIVITIES AND TASKS . . . . .	25
5.3	PROJECT PROCESSES . . . . .	26
5.3.1	PROJECT PLANNING PROCESS . . . . .	27
5.3.1.1	PURPOSE . . . . .	27
5.3.1.2	OUTCOMES . . . . .	27
5.3.1.3	ACTIVITIES AND TASKS . . . . .	27
5.3.2	PROJECT ASSESSMENT AND CONTROL PROCESS . . . . .	29
5.3.2.1	PURPOSE . . . . .	29
5.3.2.2	OUTCOMES . . . . .	29
5.3.2.3	ACTIVITIES AND TASKS . . . . .	29
5.3.3	DECISION MANAGEMENT PROCESS . . . . .	30
5.3.3.1	PURPOSE . . . . .	30
5.3.3.2	OUTCOMES . . . . .	30
5.3.3.3	ACTIVITIES AND TASKS . . . . .	30
5.3.4	RISK MANAGEMENT PROCESS . . . . .	31
5.3.4.1	PURPOSE . . . . .	31
5.3.4.2	OUTCOMES . . . . .	31
5.3.4.3	ACTIVITIES AND TASKS . . . . .	31
5.3.5	CONFIGURATION MANAGEMENT PROCESS . . . . .	33
5.3.5.1	PURPOSE . . . . .	33
5.3.5.2	OUTCOMES . . . . .	33
5.3.5.3	ACTIVITIES AND TASKS . . . . .	33
5.3.6	INFORMATION MANAGEMENT PROCESS . . . . .	34
5.3.6.1	PURPOSE . . . . .	34
5.3.6.2	OUTCOMES . . . . .	34
5.3.6.3	ACTIVITIES AND TASKS . . . . .	34
5.3.7	MEASUREMENT PROCESS . . . . .	35
5.3.7.1	PURPOSE . . . . .	35
5.3.7.2	OUTCOMES . . . . .	35
5.3.7.3	ACTIVITIES AND TASKS . . . . .	35
5.4	ORGANIZATIONAL PROCESSES . . . . .	36
5.4.1	ORGANIZATIONAL ALIGNMENT PROCESS . . . . .	37
5.4.1.1	PURPOSE . . . . .	37
5.4.1.2	OUTCOMES . . . . .	37

5.4.2	ORGANIZATION MANAGEMENT PROCESS . . . . .	38
5.4.2.1	PURPOSE . . . . .	38
5.4.2.2	OUTCOMES . . . . .	38
5.4.3	CONTRACT CHANGE MANAGEMENT PROCESS . . . . .	39
5.4.3.1	PURPOSE . . . . .	39
5.4.3.2	OUTCOMES . . . . .	39
5.5	TECHNICAL PROCESSES . . . . .	40
5.5.1	STAKEHOLDER REQUIREMENTS DEFINITION PROCESS . . . . .	41
5.5.1.1	PURPOSE . . . . .	41
5.5.1.2	OUTCOMES . . . . .	41
5.5.1.3	ACTIVITIES AND TASKS . . . . .	41
5.5.2	SYSTEM REQUIREMENTS ANALYSIS PROCESS . . . . .	43
5.5.2.1	PURPOSE . . . . .	43
5.5.2.2	OUTCOMES . . . . .	43
5.5.2.3	ACTIVITIES AND TASKS . . . . .	43
5.5.3	SYSTEM ARCHITECTURAL DESIGN PROCESS . . . . .	44
5.5.3.1	PURPOSE . . . . .	44
5.5.3.2	OUTCOMES . . . . .	44
5.5.3.3	ACTIVITIES AND TASKS . . . . .	44
5.5.4	IMPLEMENTATION PROCESS . . . . .	45
5.5.4.1	PURPOSE . . . . .	45
5.5.5	SYSTEM INTEGRATION PROCESS . . . . .	46
5.5.5.1	PURPOSE . . . . .	46
5.5.5.2	OUTCOMES . . . . .	46
5.5.5.3	ACTIVITIES AND TASKS . . . . .	46
5.5.6	SYSTEM QUALIFICATION TESTING PROCESS . . . . .	47
5.5.6.1	PURPOSE . . . . .	47
5.5.6.2	OUTCOMES . . . . .	47
5.5.6.3	ACTIVITIES AND TASKS . . . . .	47
5.5.7	SOFTWARE INSTALLATION PROCESS . . . . .	48
5.5.7.1	PURPOSE . . . . .	48
5.5.7.2	OUTCOMES . . . . .	48
5.5.7.3	ACTIVITIES AND TASKS . . . . .	48
5.5.8	SOFTWARE ACCEPTANCE SUPPORT PROCESS . . . . .	49
5.5.8.1	PURPOSE . . . . .	49
5.5.8.2	OUTCOMES . . . . .	49
5.5.8.3	ACTIVITIES AND TASKS . . . . .	49
5.5.9	SOFTWARE OPERATION PROCESS . . . . .	50
5.5.9.1	PURPOSE . . . . .	50
5.5.9.2	OUTCOMES . . . . .	50
5.5.9.3	ACTIVITIES AND TASKS . . . . .	50
5.5.10	SOFTWARE MAINTENANCE PROCESS . . . . .	52
5.5.10.1	PURPOSE . . . . .	52
5.5.10.2	OUTCOMES . . . . .	52
5.5.10.3	ACTIVITIES AND TASKS . . . . .	52

5.5.11 SOFTWARE DISPOSAL PROCESS . . . . .	54
5.5.11.1 PURPOSE . . . . .	54
5.5.11.2 OUTCOMES . . . . .	54
5.5.11.3 ACTIVITIES AND TASKS . . . . .	54
6 SOFTWARE SPECIFIC PROCESSES . . . . .	56
6.1 SOFTWARE IMPLEMENTATION PROCESSES . . . . .	56
6.1.1 SOFTWARE IMPLEMENTATION PROCESS . . . . .	57
6.1.1.1 PURPOSE . . . . .	57
6.1.1.2 OUTCOMES . . . . .	57
6.1.1.3 ACTIVITIES AND TASKS . . . . .	57
6.1.2 SOFTWARE REQUIREMENTS ANALYSIS PROCESS . . . . .	59
6.1.2.1 PURPOSE . . . . .	59
6.1.2.2 OUTCOMES . . . . .	59
6.1.2.3 ACTIVITIES AND TASKS . . . . .	59
6.1.3 SOFTWARE ARCHITECTURAL DESIGN PROCESS . . . . .	61
6.1.3.1 PURPOSE . . . . .	61
6.1.3.2 OUTCOMES . . . . .	61
6.1.3.3 ACTIVITIES AND TASKS . . . . .	61
6.1.4 SOFTWARE DETAILED DESIGN PROCESS . . . . .	62
6.1.4.1 PURPOSE . . . . .	62
6.1.4.2 OUTCOMES . . . . .	62
6.1.4.3 ACTIVITIES AND TASKS . . . . .	62
6.1.5 SOFTWARE CONSTRUCTION PROCESS . . . . .	63
6.1.5.1 PURPOSE . . . . .	63
6.1.5.2 OUTCOMES . . . . .	63
6.1.5.3 ACTIVITIES AND TASKS . . . . .	63
6.1.6 SOFTWARE INTEGRATION PROCESS . . . . .	64
6.1.6.1 PURPOSE . . . . .	64
6.1.6.2 OUTCOMES . . . . .	64
6.1.6.3 ACTIVITIES AND TASKS . . . . .	64
6.1.7 SOFTWARE QUALIFICATION TESTING PROCESS . . . . .	66
6.1.7.1 PURPOSE . . . . .	66
6.1.7.2 OUTCOMES . . . . .	66
6.1.7.3 ACTIVITIES AND TASKS . . . . .	66
6.2 SOFTWARE SUPPORT PROCESSES . . . . .	67
6.2.1 SOFTWARE DOCUMENTATION MANAGEMENT PROCESS . . . . .	68
6.2.1.1 PURPOSE . . . . .	68
6.2.1.2 OUTCOMES . . . . .	68
6.2.1.3 ACTIVITIES AND TASKS . . . . .	68
6.2.2 SOFTWARE CONFIGURATION MANAGEMENT PROCESS . . . . .	70
6.2.2.1 PURPOSE . . . . .	70
6.2.2.2 OUTCOMES . . . . .	70
6.2.2.3 ACTIVITIES AND TASKS . . . . .	70
6.2.3 SOFTWARE QUALITY ASSURANCE PROCESS . . . . .	72
6.2.3.1 PURPOSE . . . . .	72

6.2.3.2 OUTCOMES . . . . .	72
6.2.3.3 ACTIVITIES AND TASKS . . . . .	72
6.2.4 SOFTWARE VERIFICATION PROCESS . . . . .	74
6.2.4.1 PURPOSE . . . . .	74
6.2.4.2 OUTCOMES . . . . .	74
6.2.4.3 ACTIVITIES AND TASKS . . . . .	74
6.2.5 SOFTWARE VALIDATION PROCESS . . . . .	76
6.2.5.1 PURPOSE . . . . .	76
6.2.5.2 OUTCOMES . . . . .	76
6.2.5.3 ACTIVITIES AND TASKS . . . . .	76
6.2.6 SOFTWARE REVIEW PROCESS . . . . .	78
6.2.6.1 PURPOSE . . . . .	78
6.2.6.2 OUTCOMES . . . . .	78
6.2.6.3 ACTIVITIES AND TASKS . . . . .	78
6.2.7 SOFTWARE AUDIT PROCESS . . . . .	80
6.2.7.1 PURPOSE . . . . .	80
6.2.7.2 OUTCOMES . . . . .	80
6.2.7.3 ACTIVITIES AND TASKS . . . . .	80
6.2.8 SOFTWARE PROBLEM RESOLUTION PROCESS . . . . .	81
6.2.8.1 PURPOSE . . . . .	81
6.2.8.2 OUTCOMES . . . . .	81
6.2.8.3 ACTIVITIES AND TASKS . . . . .	81
6.3 SOFTWARE REUSE PROCESSES . . . . .	82
6.3.1 DOMAIN ENGINEERING PROCESS . . . . .	83
6.3.1.1 PURPOSE . . . . .	83
6.3.1.2 OUTCOMES . . . . .	83
6.3.1.3 ACTIVITIES AND TASKS . . . . .	83
6.3.2 REUSE ASSET MANAGEMENT PROCESS . . . . .	85
6.3.2.1 PURPOSE . . . . .	85
6.3.2.2 OUTCOMES . . . . .	85
6.3.2.3 ACTIVITIES AND TASKS . . . . .	85
6.3.3 REUSE PROGRAM MANAGEMENT PROCESS . . . . .	87
6.3.3.1 PURPOSE . . . . .	87
6.3.3.2 OUTCOMES . . . . .	87
6.3.3.3 ACTIVITIES AND TASKS . . . . .	87
A LOWER-LEVEL PROCESSES . . . . .	89
A.1 ACQUISITION PROCESS LOWER-LEVEL PROCESSES . . . . .	90
A.1.1 ACQUISITION PREPARATION PROCESS . . . . .	90
A.1.1.1 PURPOSE . . . . .	90
A.1.1.2 OUTCOMES . . . . .	90
A.1.2 SUPPLIER SELECTION PROCESS . . . . .	90
A.1.2.1 PURPOSE . . . . .	90
A.1.2.2 OUTCOMES . . . . .	91
A.1.3 AGREEMENT MONITORING PROCESS . . . . .	91
A.1.3.1 PURPOSE . . . . .	91

A.1.3.2 OUTCOMES . . . . .	91
A.1.4 ACQUIRER ACCEPTANCE PROCESS . . . . .	91
A.1.4.1 PURPOSE . . . . .	91
A.1.4.2 OUTCOMES . . . . .	91
A.2 SUPPLY PROCESS LOWER-LEVEL PROCESSES . . . . .	92
A.2.1 SUPPLIER TENDERING PROCESS . . . . .	92
A.2.1.1 PURPOSE . . . . .	92
A.2.1.2 OUTCOMES . . . . .	92
A.2.2 CONTRACT AGREEMENT PROCESS . . . . .	93
A.2.2.1 PURPOSE . . . . .	93
A.2.2.2 OUTCOMES . . . . .	93
A.2.3 PRODUCT AND SERVICE DELIVERY AND SUPPORT PROCESS . . . . .	93
A.2.3.1 PURPOSE . . . . .	93
A.2.3.2 OUTCOMES . . . . .	93
A.3 LIFE CYCLE MODEL MANAGEMENT PROCESS LOWER-LEVEL PROCESSES . . . . .	94
A.3.1 PROCESS ESTABLISHMENT PROCESS . . . . .	94
A.3.1.1 PURPOSE . . . . .	94
A.3.1.2 OUTCOMES . . . . .	94
A.3.2 PROCESS ASSESSMENT PROCESS . . . . .	95
A.3.2.1 PURPOSE . . . . .	95
A.3.2.2 OUTCOMES . . . . .	95
A.3.3 PROCESS IMPROVEMENT PROCESS . . . . .	95
A.3.3.1 PURPOSE . . . . .	95
A.3.3.2 OUTCOMES . . . . .	95
A.4 HUMAN RESOURCE MANAGEMENT PROCESS LOWER-LEVEL PROCESSES . . . . .	96
A.4.1 SKILL DEVELOPMENT PROCESS . . . . .	96
A.4.1.1 PURPOSE . . . . .	96
A.4.1.2 OUTCOMES . . . . .	96
A.4.2 SKILL ACQUISITION PROCESS . . . . .	96
A.4.2.1 PURPOSE . . . . .	96
A.4.2.2 OUTCOMES . . . . .	97
A.4.3 KNOWLEDGE MANAGEMENT PROCESS . . . . .	97
A.4.3.1 PURPOSE . . . . .	97
A.4.3.2 OUTCOMES . . . . .	97
A.5 SOFTWARE OPERATION PROCESS LOWER-LEVEL PROCESSES . . . . .	98
A.5.1 OPERATIONAL USE PROCESS . . . . .	98
A.5.1.1 PURPOSE . . . . .	98
A.5.1.2 OUTCOMES . . . . .	98
A.5.2 CUSTOMER SUPPORT PROCESS . . . . .	98
A.5.2.1 PURPOSE . . . . .	98
A.5.2.2 OUTCOMES . . . . .	99
B PROCESS VIEWS . . . . .	100
B.1 THE PROCESS VIEW CONCEPT . . . . .	100
B.2 PROCESS VIEWPOINT . . . . .	100

---

B.3 EXAMPLE: PROCESS VIEW FOR USABILITY . . . . .	100
B.3.0.1 PURPOSE . . . . .	101
B.3.0.2 PROCESSES, ACTIVITIES, TASKS . . . . .	101
C RELATIONSHIPS BETWEEN STANDARDS . . . . .	103
D TAILORING PROCESS . . . . .	105
D.1 PURPOSE OF THE TAILORING PROCESS . . . . .	105
D.2 OUTCOMES OF THE TAILORING PROCESS . . . . .	105
D.3 ACTIVITIES OF THE TAILORING PROCESS . . . . .	105

## 1. CHANGELOG

This document, in being heavily cited and used as a handbook and guiding force for entire organizations and the vast majority of currently active initiatives, will be updated regularly based on its usage and the need for it to be further extended, modified, refined, and so on.

The versioning of the document shall adhere to the most common form of software versioning, where the major version is followed by a decimal point and a minor version and the minor version is followed by a decimal point and a release or build version (major.minor.build). In this context, major versions shall be released when document-wide reorganization or deprecation is required, minor versions shall be released when sections or appendices are added, removed, or modified in ways that require they be merged, broken apart, or renamed, and builds shall be released when any change is made to the document itself.

Furthermore, the date of the release shall be appended to the major.minor.build form, separated by a dash. e.g. 1.0.0-08.31.2018 is the first version, first release, that occurred on August 31st, 2018. Note that this information may always be found in the bottom right-hand corner of every page of this document.

The source code for the document is written in LaTeX, and the author of the document would request it remain in that format so as to facilitate quick revision and release.

- **v1.0.0-08.31.2018:** initial release of document and exposure of proposed framework for process definition and reference modeling.

## 2. INTRODUCTION

This standard establishes a common framework for life cycle processes, with well-defined terminology, that can be referenced by businesses, organizations, and entities that create, manage, and provide software directly or indirectly, as a part of, or wholly comprising, their purchasable products and services, regardless of the industry in which the business itself is classified or categorized.

This standard applies to the acquisition of systems and software products and services, to the supply, development, operation, maintenance, and disposal of software products and the software portion of a system, whether performed internally or externally to an organization. Those aspects of system definition needed to provide the context for software products and services, and a process that can be employed for defining, controlling, and improving software life cycle processes, are provided herein. The processes may also be applied during the acquisition of a system that contains software.

This standard can be used in one or more of the following modes:

- **By an organization** - to help establish an environment of desired processes. These processes can be supported by an infrastructure of methods, procedures, techniques, tools, and trained personnel. The organization may then employ this environment to perform and manage its projects and progress systems through their life cycle stages. In this mode, this book is used to assess conformance of a declared, established set of life cycle processes to its provisions.
- **By a project** - to help select, structure, and employ the elements of an established set of life cycle processes to provide products and services. In this mode, this book is used in the assessment of conformance of the project to the declared and established environment.
- **By an acquirer and a supplier** - to help develop an agreement concerning processes and activities. Via the agreement, the processes and activities in this book are selected, negotiated, agreed to, and performed. In this mode, this book is used for guidance in developing the agreement.
- **By organizations and assessors** - to perform assessments that may be used to support organizational process improvement.

### 3. UNDERSTANDING AND APPLYING THIS STANDARD

#### 3.1 RELATIONSHIP BETWEEN SYSTEMS AND SOFTWARE

This standard establishes a strong link between a system and its software. It is based upon the general principles of systems engineering. 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 design, producing the software, and integrating it into the system.

It is a fundamental premise that software always exists in the context of a system, even if the system consists of only the processor upon which the software is executed. Therefore, a software product or service is always treated as one item in a system.

For example, this standard makes a distinction between system requirements analysis and software requirements analysis, because, in the general case, system architectural design will allocate the system requirements to various items of the system and software requirements analysis will derive software requirements from the system requirements allocated to each software item.

Note that in some cases, the non-software items of a system may be so minimal that it may not be necessary to perform distinct system and software analyses.

#### 3.2 ORGANIZATION-LEVEL AND PROJECT-LEVEL ADOPTION

Modern software businesses strive to develop an exhaustive set of software life cycle processes that are applied repeatedly to the software projects of the business. Therefore, this standard is intended to be useful for adoption at either the organization level or at the project level.

An organization would adopt this standard and supplement it with appropriate procedures, practices, tools and policies.

A software project of the organization would conform to the organization's processes rather than conform directly to this standard.

In some cases, projects may be executed by an organization that does not have an appropriate set of processes adopted at the organizational level. Such a project may apply the provisions of this standard directly to the project itself.

#### 3.3 TEMPORAL RELATIONSHIPS AMONG PROCESSES

Within this standard, the processes and their activities and tasks are arranged in a sequence suitable for exposition. This positional sequence does not prescribe or dictate any time-dependent sequence.

For lack of consensus on or use of a universal time-dependent sequence, the reader may select and order the processes, activities, and tasks as appropriate and effective.

This standard encourages iteration among the activities and recursion within an activity to offset the effects of any implied sequence of activities and tasks.

The organization is ultimately responsible for selecting a life cycle model for the project and mapping the processes, activities, and tasks onto that model, as well as for selecting life cycle models and processes for its own internal management and governance belonging to, or wholly separate from, those used for software projects.

### 3.4 EVALUATION VS VERIFICATION AND VALIDATION

Organizations that are involved in any process of a life cycle conduct evaluations of the products of that task. The Software Verification and Software Validation processes provide the opportunity for additional evaluations. These processes are conducted by the acquirer, the supplier, or an independent party to verify and validate the products in varying depth depending on the project.

These evaluations do not duplicate or replace other evaluations, but supplement them. Additional opportunities for evaluation are provided by the Software Review, Software Audit, Software Quality Assurance and the Life Cycle Model Management Processes.

### 3.5 CRITERIA FOR PROCESSES

This standard establishes a framework for the life cycle of software. The life cycle begins with an idea or a need that can be satisfied wholly or partly by software and ends with the retirement of the software. The architecture is built with a set of processes and interrelationships among these processes. The determination of the life cycle processes is based upon two basic principles: cohesion and responsibility.

- **Cohesion:** The life cycle processes are cohesive and coupled to the optimum extent deemed practical and feasible
- **Responsibility:** A process is placed under the responsibility of an organization or role in the software life cycle

### 3.6 DESCRIPTION AND GENERAL CHARACTERISTICS OF PROCESSES

The processes within this standard are described in a manner that is meant to facilitate the use of this standard for either, or both, organization-level and project-level adoption. Each process is described in terms of the following attributes:

- **Title:** The title conveys the scope of the process as a whole
- **Purpose:** The purpose describes the goals of performing the process
- **Outcomes:** The outcomes express the observable results expected from the successful performance of the process
- **Activities:** The activities are a list of actions that are used to achieve the outcomes
- **Tasks:** The tasks are requirements, recommendations, or permissible actions intended to support the achievement of the outcomes

### 3.7 DECOMPOSITION OF PROCESSES

Each process within this standard satisfies the criteria described above. For the purpose of clear description, processes are sometimes decomposed into smaller pieces. Some processes are decomposed into activities and/or lower-level processes. A lower-level process is described when the decomposed portion of the process itself satisfies the criteria to be a process. An activity is used when the decomposed unit does not qualify as a process. An activity can be considered as simply a collection of tasks.

A task is expressed in the form of a requirement, recommendation, or permissible action, intended to support the achievement of the outcomes of a process. For this purpose, this standard carefully employs certain auxiliary verbs (shall, should, and may) to differentiate between the distinct forms of a task. “Shall” is used to express a provision required for conformance, “should” to express a recommendation among other possibilities, and “may” to indicate a course of action permissible within the limits of this standard and common framework.

### 3.8 LIFE CYCLE MODELS AND STAGES

The life of a system or a software product can be modeled by a life cycle model consisting of stages. Models may be used to represent the entire life from concept to disposal or to represent the portion of the life corresponding to the current project. The life cycle model is comprised of a sequence of stages that may overlap and/or iterate, as per the project’s scope, magnitude, complexity, changing needs and opportunities.

Each stage is described with a statement of purpose and outcomes. The life cycle processes and activities are selected and employed in a stage to fulfill the purpose and outcomes of that stage.

Different organizations may undertake different stages in the life cycle. However, each stage is conducted by the organization responsible for that stage with due consideration of the available information on life cycle plans and decisions made in preceding stages. Similarly, the organization responsible for that stage records the decisions made and records the assumptions regarding subsequent stages in the life cycle.

Finally, this standard does not dictate the use of any specific combination or permutation of stages to explicitly and always define any given life cycle. It is left to the organization to determine, by using this standard and its examples and exhaustive breakdown and definition of life cycle processes, those life cycles beneath their own business operations and processes, their own tactical initiatives, activities, and tasks.

## 4. STRUCTURE, ORGANIZATION, AND IMPLEMENTATION

### 4.1 CATEGORIES OF LIFE CYCLE PROCESSES

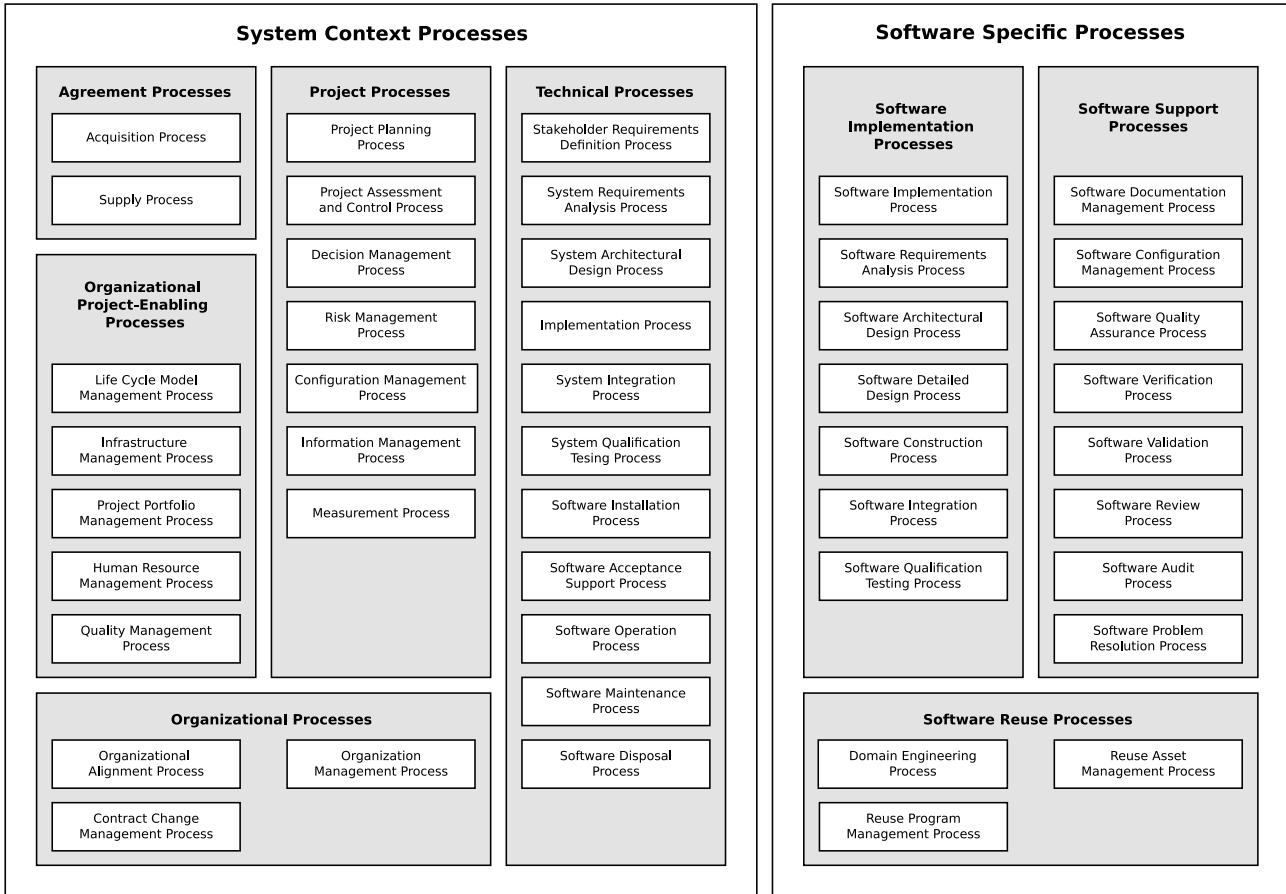


Figure 1: Process Reference Model (PRM)

To facilitate practical understanding and application of this standard, we have grouped the activities that may be performed during the life cycle of a software system into seven process groups. Each of the life cycle processes within those groups is described in terms of its purpose and desired outcomes, and lists activities and tasks which need to be performed to achieve those outcomes.

The purpose and outcomes of the life cycle processes constitute a Process Reference Model (PRM), where each process group is contained within its own distinct section, the process groups and their processes are sub-sections within those sections, and each sub-section / process contains sub-sub-sections for their purpose, outcome, activities, and tasks.

### 4.2 SUMMARY OF LIFE CYCLE PROCESSES

There are two major sub-divisions of process within this standard, where Section SYSTEM CONTEXT PROCESSES provides a system context for dealing with a standalone software product or service of a software system, and Section SOFTWARE SPECIFIC PROCESSES contains the software-specific processes for use in implementing a software product or service that is an element of a larger system. Brief summarization of these divisions' process groups may be found below.

#### 4.2.1 AGREEMENT PROCESSES

AGREEMENT PROCESSES define the activities necessary to establish an agreement between two organizations.

#### 4.2.2 ORGANIZATIONAL PROJECT-ENABLING PROCESSES

The ORGANIZATIONAL PROJECT-ENABLING PROCESSES manage the organization's capability to acquire and supply products or services through the initiation, support and control of projects. They provide resources and infrastructure necessary to support projects and ensure the satisfaction of organizational objectives and established agreements.

#### 4.2.3 PROJECT PROCESSES

There are two categories of PROJECT PROCESSES. The Project Management Processes are used to plan, execute, assess and control the progress of a project. The Project Support Processes support specialized management objectives.

#### 4.2.4 ORGANIZATIONAL PROCESSES

The ORGANIZATIONAL PROCESSES are highly useful, in that they help organizations ensure their processes are aligned and consistent with its business goals.

#### 4.2.5 TECHNICAL PROCESSES

The TECHNICAL PROCESSES are used to define the requirements for a system, to transform the requirements into an effective product, to permit consistent reproduction of the product where necessary, to use the product, to provide the required services, to sustain the provision of those services and to dispose of the product when it is retired from service.

#### 4.2.6 SOFTWARE IMPLEMENTATION PROCESSES

The SOFTWARE IMPLEMENTATION PROCESSES are used to produce a specified system element (software item) implemented in software.

Those processes transform specified behavior, interfaces and implementation constraints into implementation actions resulting in a system element that satisfies the requirements derived from the system requirements.

#### 4.2.7 SOFTWARE SUPPORT PROCESSES

The SOFTWARE SUPPORT PROCESSES provide a specific focused set of activities for performing a specialized software process. A supporting process assists the Software Implementation Process as an integral part with a distinct purpose, contributing to the success and quality of the software project.

#### 4.2.8 SOFTWARE REUSE PROCESSES

The SOFTWARE REUSE PROCESSES consists of three processes that support an organization's ability to reuse software items across project boundaries. These processes are unique because, by their nature, they operate outside the bounds of any particular project.

#### 4.3 STANDARD AS PROCESS REFERENCE MODEL (PRM)

A Process Reference Model (PRM) provides a level of abstraction higher than that of the detailed required contained with any given standard or specification, and is applicable to an organization that is assessing its processes in order to determine their own capabilities and limitations.

A Process Reference Model, by definition, must contain the following:

1. A declaration of the target domain
2. A description of the processes within its scope
3. A description of the relationship between itself and its intended context of use
4. A description of the relationship between the processes defined within itself

A Process Reference Model does not necessarily represent a particular process implementation approach, nor does it prescribe a system/software life cycle model, methodology, or technique. Instead, the reference model is intended to be adopted by an organization based on its business needs and application domain.

Furthermore, the organization's defined process is adopted by the organization's projects in the context of the projects' requirements.

In order to ensure this standard serves as a legitimate Process Reference Model, it has been written and structured in a manner and method that satisfies international standards and guidelines while also, hopefully, being informative, educational, and useful to its readers.

## 5. SYSTEM CONTEXT PROCESSES

### 5.1 AGREEMENT PROCESSES

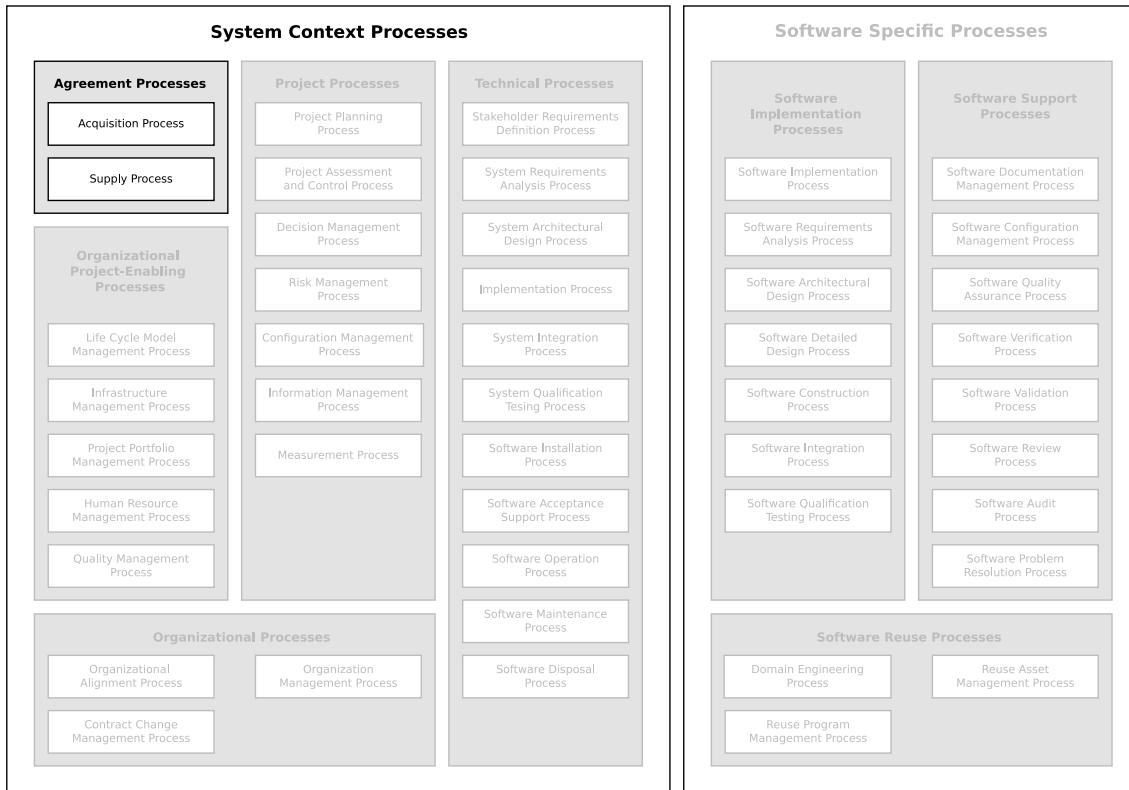


Figure 2: Agreement Processes

If the ACQUISITION PROCESS is invoked, it provides the means for conducting business with a supplier of products that are supplied for use as an operational system, of services in support of an operational system, or of elements of a system being developed by a project.

If the SUPPLY PROCESS is invoked, it provides the means for conducting a project in which the result is a product or service that is delivered to the acquirer.

In this Section:

- 5.1.1 - ACQUISITION PROCESS
- 5.1.2 - SUPPLY PROCESS

## 5.1.1 ACQUISITION PROCESS

### 5.1.1.1 PURPOSE

The purpose of the Acquisition Process is to obtain the product and/or service that satisfies the need expressed by the acquirer. The process begins with the identification of customer needs and ends with the acceptance of the product and/or service needed by the acquirer.

### 5.1.1.2 OUTCOMES

- acquisition needs, goals, product and/or service acceptance criteria and acquisition strategies are defined;
- an agreement is developed that clearly expresses the expectation, responsibilities and liabilities of both the acquirer and the supplier;
- one or more suppliers is selected;
- a product and/or service is acquired that satisfies the acquirer's stated need
- the acquisition is monitored so that specified constraints such as cost, schedule and quality are met;
- supplier deliverables are accepted; and
- any identified open items have a satisfactory conclusion as agreed to by the acquirer and the supplier.

### 5.1.1.3 ACTIVITIES AND TASKS

#### 1. Acquisition Preparation

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.
2. The acquirer shall define and analyze the system requirements. The system requirements should include business, organizational and user as well as safety, security, and other criticality requirements along with related design, testing, and compliance standards and procedures.
3. The acquirer may perform the definition and analysis of software requirements by itself or may retain a supplier to perform this task.
4. If the acquirer retains a supplier to perform system or software requirements analysis, the acquirer shall retain approval authority for the analyzed requirements.
5. The Technical Processes should be used to perform the tasks above. The acquirer may use the Stakeholder Requirements Definition Process to establish the customer requirements.
6. The acquirer shall consider options for acquisition against analysis of appropriate criteria to include risk, cost and benefits for each option. Options include:
  - Purchase an off-the-shelf software product that satisfies the requirements.
  - Develop the software product or obtain the software service internally.
  - Develop the software product or obtain the software service through contract.
  - A combination of the three items above.
  - Enhance an existing software product or service.
7. When an off-the-shelf software product is to be acquired, the acquirer shall ensure the following conditions are satisfied:
  - The requirements for the software product are satisfied.
  - The required documentation is available.

- Proprietary, usage, ownership, warranty and licensing rights are satisfied.
  - Future support for the software product is planned.
8. The acquirer should prepare, document and execute an acquisition plan. The plan should contain the following:
- Requirements for the system.
  - Planned employment of the system.
  - Type of contract to be employed.
  - Responsibilities of the organizations involved.
  - Support concept to be used.
  - Risks considered as well as methods to manage the risks.
9. The acquirer shall define and document the acceptance strategy and conditions (criteria).
10. The acquirer should document the acquisition requirements (e.g., request for proposal), the content of which depends upon the acquisition option selected in previously defined task. The acquisition documentation should include, as appropriate:
- System requirements.
  - Scope statement.
  - Instructions for bidders.
  - List of software products.
  - Terms and conditions.
  - Control of subcontracts.
  - Technical constraints (e.g. target environment).
11. The acquirer should determine which processes of this standard are appropriate for the acquisition and specify any acquirer requirements for tailoring those processes. The acquirer should specify if any of the processes are to be performed by parties other than the supplier, so that suppliers may, in their proposals, define their approach to supporting the work of other parties. The acquirer shall define the scope of those tasks that reference the contract.
12. The acquisition documentation shall also define the contract milestones at which the supplier's progress shall be reviewed and audited as part of monitoring the acquisition.
13. The acquisition requirements should be given to the organization selected for performing the acquisition activities.
2. **Acquisition Advertisement**
1. The acquirer shall communicate the request for the supply of a product or service to identified suppliers.
3. **Supplier Selection**
1. The acquirer should establish a procedure for supplier selection including proposal evaluation criteria and requirements compliance weighting.
  2. The acquirer should select a supplier based upon the evaluation of the suppliers' proposals, capabilities, and in accordance with the acquirer's acceptance strategy and conditions.
4. **Contract Agreement**
1. The acquirer may involve other parties, including potential suppliers or any necessary third parties (such as regulators), before contract award, in determining the acquirer's requirements for tailoring of this standard for the project. In making this determination, the acquirer shall consider the effect of the tailoring requirements upon the supplier's organizationally-adopted processes. The acquirer shall include or reference the tailoring requirements in the contract.
  2. The acquirer shall 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 shall address proprietary, usage, ownership, warranty and licensing rights associated with the reusable off-the-shelf software products.

3. Once the contract is underway, the acquirer shall 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.
5. **Agreement Monitoring**
  1. The acquirer shall monitor the supplier's activities in accordance with the Software Review Process and the Software Audit Process. The acquirer should supplement the monitoring with the Software Verification Process and the Software Validation Process as needed.
  2. The acquirer shall cooperate with the supplier to provide all necessary information in a timely manner and resolve all pending items.
6. **Acquirer Acceptance**
  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.
  2. The acquirer shall conduct acceptance review and acceptance testing of the deliverable software product or service and shall accept it from the supplier when all acceptance conditions are satisfied. The acceptance procedure should comply with the provisions defined by previous activities and tasks within this process.
  3. After acceptance, the acquirer should take the responsibility for the configuration management of the delivered software product
7. **Closure**
  1. The acquirer shall make payment or provide other agreed consideration to the supplier for the product or service rendered.

## 5.1.2 SUPPLY PROCESS

### 5.1.2.1 PURPOSE

The purpose of the Supply Process is to provide a product or service to the acquirer that meets the agreed requirements.

### 5.1.2.2 OUTCOMES

- An acquirer for a product or service is identified;
- A response to an acquirer's request is produced;
- An agreement is established between the acquirer and the supplier for developing, maintaining, operating, packaging, delivering, and installing the product and/or service;
- A product and/or service that meets the agreed requirements are developed by the supplier;
- The product and/or service is delivered to the acquirer in accordance with the agreed requirements; and
- The product is installed in accordance with the agreed requirements.

### 5.1.2.3 ACTIVITIES AND TASKS

#### 1. Opportunity Identification

1. The supplier should determine the existence and identity of an acquirer who has, or who represents an organization or organizations having, a need for a product or service.

#### 2. Supplier Tendering

1. The supplier should conduct a review of requirements in the request for proposal taking into account organizational policies and other regulations.
2. The supplier should make a decision to bid or accept the contract.
3. The supplier shall prepare a proposal in response to the request for proposal.

#### 3. Contract Agreement

1. The supplier shall negotiate and enter into a contract with the acquirer to provide the software product or service.
2. The supplier may request modification to the contract as part of the change control mechanism.

#### 4. Contract Execution

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.
2. If not stipulated in the contract, the supplier shall define or select a life cycle model appropriate to the scope, magnitude, and complexity of the project. The life cycle model shall be comprised of stages and the purpose and outcomes of each stage. The processes, activities, and tasks of this standard shall be selected and mapped onto the life cycle model.
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.
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:

1. Develop the software product or provide the software service using internal resources.
2. Develop the software product or provide the software service by subcontracting.
3. Obtain off-the-shelf software products from internal or external sources.
4. A combination of a, b, and c above.
5. The supplier shall develop and document project management plan(s) based upon the planning requirements and options selected in previous point
  1. Project organizational structure and authority and responsibility of each organizational unit, including external organizations.
  2. Engineering environment (for development, operation, or maintenance, as applicable), including test environment, library, equipment, facilities, standards, procedures, and tools.
  3. Work breakdown structure of the life cycle processes and activities, including the software products, software services and non-deliverable items, to be performed together with budgets, staffing, physical resources, software size, and schedules associated with the tasks.
  4. Management of the quality characteristics of the software products or services. Separate plans for quality may be developed.
  5. Management of the safety, security, and other critical requirements of the software products or services. Separate plans for safety and security may be developed.
  6. Subcontractor management, including subcontractor and the acquirer, if any. subcontractor selection and involvement between the subcontractor and the acquirer, if any.
  7. Quality assurance.
  8. Verification and validation, including the approach for interfacing with the verification and validation agent, if specified.
  9. Acquirer involvement; that is, by such means as reviews, audits, informal meetings, reporting, modification and change; implementation, approval, acceptance, and access to facilities.
  10. User involvement; by such means as requirements setting exercises, prototype demonstrations and evaluations.
  11. Risk management; that is, management of the areas of the project that involve potential technical, cost, or schedule risks.
  12. Security policy; that is, the rules for need-to-know and access-to-information at each project organization level.
  13. Approval required by such means as regulations, required certifications, proprietary, usage, ownership, warranty and licensing rights.
  14. Means for scheduling, tracking, and reporting.
  15. Training of personnel.
6. The supplier shall implement and execute the project management plan(s) developed in previous point.
7. The supplier shall:
  1. Develop the software product in accordance with the Technical Processes
  2. Operate the software product in accordance with the Software Operation Process
  3. Maintain the software product in accordance with the Software Maintenance Process
8. The supplier shall monitor and control the progress and the 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:
  1. Monitoring progress of technical performance, costs, and schedules and reporting of project

status.

2. Problem identification, recording, analysis, and resolution.
  9. The supplier shall manage and control the subcontractors in accordance with the Acquisition Process. 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.
  10. The supplier shall interface with the independent verification, validation, or test agent as specified in the contract and project plans.
  11. The supplier shall interface with other parties as specified in the contract and project plans.
  12. The supplier should coordinate contract review activities, interfaces, and communication with the acquirer's organization.
  13. The supplier shall conduct or support the informal meetings, acceptance review, acceptance testing, joint reviews, and audits with the acquirer as specified in the contract and project plans. The joint reviews shall be conducted in accordance with the SOFTWARE REVIEW PROCESS, audits in accordance with the SOFTWARE AUDIT PROCESS.
  14. The supplier should perform verification and validation in accordance with the SOFTWARE VERIFICATION PROCESS and SOFTWARE VALIDATION PROCESS respectively to demonstrate that the software products or services and processes fully satisfy their respective requirements.
  15. The supplier shall make available to the acquirer the reports of evaluation, reviews, audits, testing, and problem resolutions as specified in the contract.
  16. 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.
  17. The supplier shall perform quality assurance activities in accordance with the SOFTWARE QUALITY ASSURANCE PROCESS.
5. **Product/Service Delivery and Support**
1. The supplier shall deliver the software product or service as specified in the contract.
  2. The supplier shall provide assistance to the acquirer in support of the delivered software product or service as specified in the contract.
6. **Closure:**
1. The supplier shall accept and acknowledge payment or other agreed consideration.
  2. The supplier shall transfer the responsibility for the product or service to the acquirer, or other party, as directed by the agreement.

## 5.2 ORGANIZATIONAL PROJECT-ENABLING PROCESSES

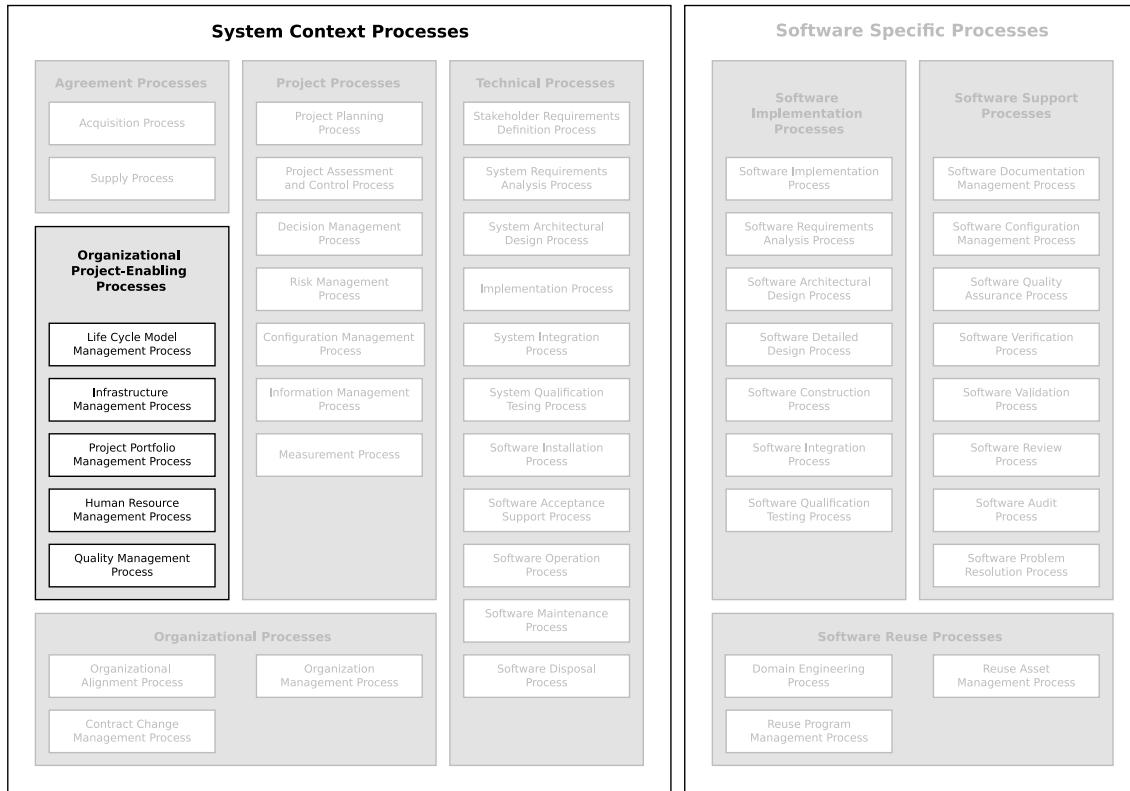


Figure 3: Organizational Project-Enabling Processes

The ORGANIZATIONAL PROJECT-ENABLING PROCESSES manage the organization's capability to acquire and supply products or services through the initiation, support and control of projects.

They provide resources and infrastructure necessary to support projects and ensure the satisfaction of organizational objectives and established agreements. They are not intended to be a comprehensive set of business processes that enable management of the organization's business.

In this Section:

- 5.2.1 - LIFE CYCLE MODEL MANAGEMENT PROCESS
  - 5.2.2 - INFRASTRUCTURE MANAGEMENT PROCESS
  - 5.2.3 - PROJECT PORTFOLIO MANAGEMENT PROCESS
  - 5.2.4 - HUMAN RESOURCE MANAGEMENT PROCESS
  - 5.2.5 - QUALITY MANAGEMENT PROCESS

## 5.2.1 LIFE CYCLE MODEL MANAGEMENT PROCESS

### 5.2.1.1 PURPOSE

The purpose of the Life Cycle Model Management Process is to define, maintain, and assure availability of policies, life cycle processes, life cycle models, and procedures for use by the organization with respect to the scope of this standard.

This process provides life cycle policies, processes, and procedures that are consistent with the organization's objectives, that are defined, adapted, improved and maintained to support individual project needs within the context of the organization, and that are capable of being applied using effective, proven methods and tools.

### 5.2.1.2 OUTCOMES

As a result of the successful implementation of the Life Cycle Model Management Process:

- policies and procedures for the management and deployment of life cycle models and processes are provided;
- responsibility, accountability and authority for life cycle management are defined;
- life cycle processes, models and procedures for use by the organization are defined, maintained and improved; and
- prioritized process improvements are implemented.

### 5.2.1.3 ACTIVITIES AND TASKS

#### 1. Process establishment

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

#### 2. Process assessment

1. The organization should develop, document and apply a process assessment procedure. Assessment records should be produced and maintained.
2. The organization shall plan and carry out reviews of the processes at appropriate intervals to ensure their continuing suitability and effectiveness in the light of assessment results.

#### 3. Process improvement

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.
2. Historical, technical, and evaluation data should be collected and analyzed to gain an understanding of the strengths and weaknesses of the employed processes. These analyses should be used as feedback to improve these processes, to recommend changes in the direction of the projects (or subsequent projects), and to determine technology advancement needs.
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.

## 5.2.2 INFRASTRUCTURE MANAGEMENT PROCESS

### 5.2.2.1 PURPOSE

The purpose of the Infrastructure Management Process is to provide the enabling infrastructure and services to projects to support organization and project objectives throughout the life cycle.

This process defines, provides and maintains the facilities, tools, and communications and information technology assets needed for the organization's business with respect to the scope of this standard.

### 5.2.2.2 OUTCOMES

- The requirements for infrastructure to support processes are defined;
- The infrastructure elements are identified and specified;
- The infrastructure elements are implemented; and
- A stable and reliable infrastructure is maintained and improved.

**Note:** The infrastructure elements may include hardware, software, methods, tools, techniques, standards, and facilities for development, operation, or maintenance.

### 5.2.2.3 ACTIVITIES AND TASKS

The organization shall implement the following activities in accordance with applicable organization policies and procedures with respect to the INFRASTRUCTURE MANAGEMENT PROCESS .

#### 1. Process implementation:

1. The infrastructure should be defined and documented to meet the requirements of the process employing this process, considering the applicable procedures, standards, tools, and techniques.
2. The establishment of the infrastructure should be planned and documented.

#### 2. Establishment of the infrastructure:

1. The configuration of the infrastructure should be planned and documented. Functionality, performance, safety, security, availability, space requirements, equipment, costs, and time constraints should be considered.
2. The infrastructure shall be installed in time for execution of the relevant process.

#### 3. Maintenance of the infrastructure:

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

### 5.2.3 PROJECT PORTFOLIO MANAGEMENT PROCESS

#### 5.2.3.1 PURPOSE

The purpose of the PROJECT PORTFOLIO MANAGEMENT PROCESS is to initiate and sustain necessary, sufficient and suitable projects in order to meet the strategic objectives of the organization.

This process commits the investment of adequate organization funding and resources, and sanctions the authorities needed to establish selected projects. It performs continued qualification of projects to confirm they justify, or can be redirected to justify, continued investment.

#### 5.2.3.2 OUTCOMES

- Business venture opportunities, investments or necessities are qualified, prioritized and selected;
- Resources and budgets for each project are identified and allocated;
- Project management accountability and authorities are defined;
- Projects meeting agreement and stakeholder requirements are sustained; and
- projects not meeting agreement or stakeholder requirements are redirected or terminated.

#### 5.2.3.3 ACTIVITIES AND TASKS

The organization shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the PROJECT PORTFOLIO MANAGEMENT PROCESS.

##### 1. Project Initiation:

1. The organization shall identify, prioritize, select, and establish new business opportunities, ventures, or undertakings in a manner that is consistent with the business strategy and action plans of the organization.
2. The organization shall define accountabilities and authorities for each project.
3. The organization shall identify the expected outcomes of the projects.
4. The organization shall allocate resources for the achievement of project objectives.
5. The organization shall identify any multi-project interfaces that must be managed or supported by the project.
6. The organization shall specify the project reporting requirements and review milestones that will govern the execution of the project.
7. The organization shall authorize the project to commence execution of approved project plans, including the technical plans.

##### 2. Project Evaluation:

1. The organization shall evaluate ongoing projects to confirm that:
  1. Projects are making progress towards achieving established goals.
  2. Projects are complying with project directives.
  3. Projects are being conducted according to system life cycle plans and procedures.
  4. Projects remain viable, as indicated by, for example, continuing need for the service, practicable product implementation, acceptable investment benefits.
2. The organization shall act to continue or redirect projects that are satisfactorily progressing or can be expected to progress satisfactorily by appropriate redirection

##### 3. Project Closure:

1. The organization shall act to cancel or suspend projects whose disadvantages or risks to the organization outweigh the benefits of continued investments, where agreements permit this.
2. After completion of the agreement for products and services, the organization shall act to close the project per organizational policies and procedures and the agreement.

## 5.2.4 HUMAN RESOURCE MANAGEMENT PROCESS

### 5.2.4.1 PURPOSE

The purpose of the HUMAN RESOURCE MANAGEMENT PROCESS is to provide the organization with necessary human resources and to maintain their competencies, consistent with business needs.

The process assures the providing of a supply of skilled and experienced personnel qualified to perform life cycle processes to achieve organization, project and customer objectives.

### 5.2.4.2 OUTCOMES

- skills required by projects are identified;
- necessary human resources are provided to projects;
- skills of personnel are developed, maintained or enhanced;
- conflicts in multi-project resource demands are resolved; and
- individual knowledge, information and skills are collected, shared, reused and improved throughout the organization.

### 5.2.4.3 ACTIVITIES AND TASKS

#### 1. Skill Identification

1. A review of the organization and 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. These needs may be met through training, recruitment or other staff development mechanisms.
2. The types and levels of training and knowledge needed to satisfy organization and project requirements shall be determined.

#### 2. Skill Development

1. A training plan, addressing implementation schedules, resource requirements, and training needs, should be developed and documented.
2. Training manuals, including presentation materials used in providing training should be developed or acquired.
3. The training plan shall be implemented to provide training to personnel. Training records should be maintained.

#### 3. Skill Acquisition and Provision

1. Establish a systematic program for recruitment of staff qualified to meet the needs of the organization and projects. Provide opportunities for the career development of existing staff.
2. Define objective criteria that can be used to evaluate staff performance.
3. Evaluate the performance of the staff in respect of their contributions to the goals of the organization or project.
4. Ensure that feedback is provided to the staff on the results of any evaluations performed.
5. Maintain adequate records of staff performance including information on skills, training completed, and performance evaluations.
6. Define the organization's and project's need for project teams. Define team structure and operating rules.

7. Conflicts in multi-project resource demands should be resolved.
  8. An understanding of their role on the project.
  9. A shared vision or sense of common interests on the success of the project.
  10. Appropriate mechanisms or facilities for communication and interactions among teams.
  11. Support from appropriate management to accomplish project requirements.
  12. 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.
4. **Knowledge Management**
1. The organization shall plan the requirements for managing the organization's knowledge assets. The planning shall include the definition of the infrastructure and training to support the contributors and the users of the organization's knowledge assets, the classification schema for the assets and the asset criteria.
  2. The organization shall establish a network of experts within the organization. The network shall contain the identification of the organization's experts, a list of their area of expertise and the identification of available information within a classification schema, e.g., knowledge area. The organization shall ensure that the network is maintained current.
  3. The organization shall establish a mechanism to support the exchange of information between the experts and the flow of expert information to the organization's projects. The mechanism shall support the organization's access, storage and retrieval requirements.
  4. The organization shall perform configuration management of assets in accordance with the CONFIGURATION MANAGEMENT PROCESS.
  5. The organizations shall capture and maintain information for access by the organization per the plan.

## 5.2.5 QUALITY MANAGEMENT PROCESS

### 5.2.5.1 PURPOSE

The purpose of the QUALITY MANAGEMENT PROCESS is to assure that products, services and implementations of life cycle processes meet organizational quality objectives and achieve customer satisfaction.

### 5.2.5.2 OUTCOMES

- organization quality management policies and procedures are defined;
- organization quality objectives are defined;
- accountability and authority for quality management are defined;
- the status of customer satisfaction is monitored; and
- appropriate action is taken when quality objectives are not achieved.

### 5.2.5.3 ACTIVITIES AND TASKS

#### 1. Quality Management

1. The organization shall establish quality management policies, standards and procedures.
2. The organization shall establish organization quality management goals and objectives based on business strategy for customer satisfaction.
3. The organization shall define responsibilities and authority for implementation of quality management.
4. The organization shall assess customer satisfaction and report.
5. The organization shall conduct periodic reviews of project quality plans.
6. The organization shall monitor the status of quality improvements on products and services.

#### 2. Quality Management Corrective Action

1. The organization shall take corrective actions when quality management goals are not met.
2. The organization shall implement corrective actions and communicate results through the organization.

### 5.3 PROJECT PROCESSES

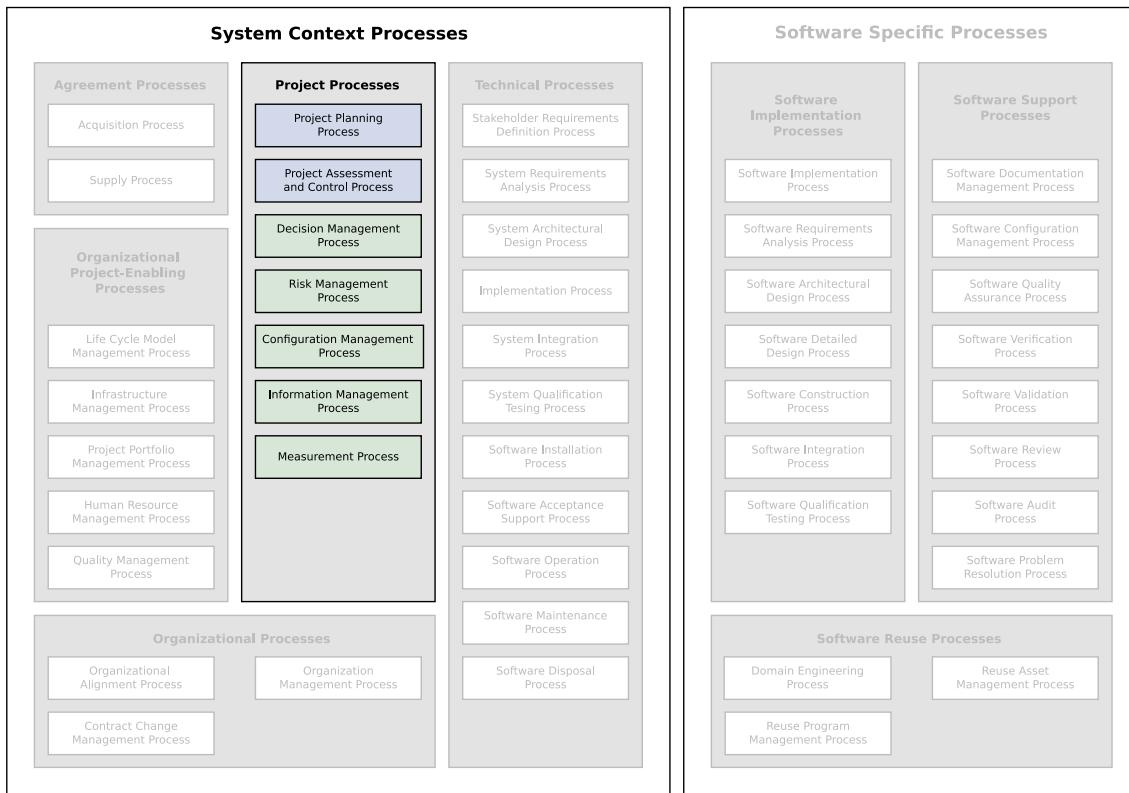


Figure 4: Project Processes

Within this standard, the project has been chosen as the context for describing processes concerned with planning, assessment and control. The principles related to these processes can be applied in any area of an organization's management.

There are two categories of Project Processes. The Project Management Processes are used to plan, execute, assess and control the progress of a project. The Project Support Processes support specialized management objectives. Both are described below.

The Project Management Processes are used to establish and evolve project plans, to assess actual achievement and progress against the plans and to control execution of the project through to fulfillment. Individual Project Management Processes may be invoked at any time in the life cycle and at any level in a hierarchy of projects, as required by project plans or unforeseen events. The Project Management Processes are applied with a level of rigor and formality that depends on the risk and complexity of the project.

The Project Support Processes provide a specific focused set of tasks for performing a specialized management objective. They are all evident in the management of any undertaking, ranging from a complete organization down to a single life cycle process and its tasks.

### 5.3.1 PROJECT PLANNING PROCESS

#### 5.3.1.1 PURPOSE

The purpose of the PROJECT PLANNING PROCESS is to produce and communicate effective and workable project plans.

This process determines the scope of the project management and technical activities, identifies process outputs, project tasks and deliverables, establishes schedules for project task conduct, including achievement criteria, and required resources to accomplish project tasks.

#### 5.3.1.2 OUTCOMES

- The scope of the work for the project is defined;
- The feasibility of achieving the goals of the project with available resources and constraints are evaluated;
- The tasks and resources necessary to complete the work are sized and estimated;
- Interfaces between elements in the project, and with other project and organizational units, are identified;
- Plans for the execution of the project are developed; and
- Plans for the execution of the project are activated.

#### 5.3.1.3 ACTIVITIES AND TASKS

##### 1. Project initiation:

1. The manager shall establish the requirements of the project to be undertaken.
2. Once the project requirements are established, the manager shall establish the feasibility of the project by checking that the resources (personnel, materials, technology, and environment) required to execute and manage the project are available, adequate, and appropriate and that the timescales to completion are achievable.
3. As necessary, and by agreement of all parties concerned, the requirements of the project may be modified at this point to achieve the completion criteria.

##### 2. Project planning:

1. The manager shall prepare the plans for execution of the project. The plans associated with the execution of the project shall contain descriptions of the associated activities and tasks and identification of the software products that will be provided. These plans shall include, but are not limited to, the following:
  1. Schedules for the timely completion of tasks.
  2. Estimation of effort.
  3. Adequate resources needed to execute the tasks.
  4. Allocation of tasks.
  5. Assignment of responsibilities.
  6. Quantification of risks associated with the tasks or the process itself.
  7. Quality assurance measures to be employed throughout the project.
  8. Costs associated with the process execution.
  9. Provision of environment and infrastructure.

10. Definition and maintenance of a life cycle model that is comprised of stages using the defined life cycle models for projects of the organization
3. **Project activation:**
1. The manager shall obtain authorization for the project.
  2. The manager shall submit requests for necessary resources to perform the project.
  3. The manager shall initiate the implementation of the project plan/s to satisfy the objectives and criteria set, exercising control over the project

### 5.3.2 PROJECT ASSESSMENT AND CONTROL PROCESS

#### 5.3.2.1 PURPOSE

The purpose of the PROJECT ASSESSMENT AND CONTROL PROCESS is to determine the status of the project and ensure that the project performs according to plans and schedules, and within projected budgets, and that it satisfies technical objectives.

This process includes redirecting the project activities, as appropriate, to correct identified deviations and variations from other project management or technical processes. Redirection may include replanning as appropriate.

#### 5.3.2.2 OUTCOMES

- progress of the project is monitored and reported;
- interfaces between elements in the project, and with other project and organizational units, are monitored;
- actions to correct deviations from the plan and to prevent recurrence of problems identified in the project are taken when project targets are not achieved; and
- project objectives are achieved and recorded.

#### 5.3.2.3 ACTIVITIES AND TASKS

##### 1. Project Monitoring

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

##### 2. Project Control

1. The manager shall investigate, analyze, and resolve the problems discovered during the execution of the project. 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.
2. The manager shall report, at agreed points, the progress of the project, 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.

##### 3. Project Assessment

1. The manager shall ensure that the software products and plans are evaluated for satisfaction of requirements.
2. The manager shall assess the evaluation results of the software products, activities, and tasks completed during the execution of the project for achievement of the objectives and completion of the plans.

##### 4. Project Closure

1. When all software products, activities, and tasks are completed, the manager shall determine whether the project is complete, taking into account the criteria as specified in the contract or as part of organization's procedure.
2. These results and records shall be archived in a suitable environment as specified in the contract.

### 5.3.3 DECISION MANAGEMENT PROCESS

#### 5.3.3.1 PURPOSE

The purpose of the DECISION MANAGEMENT PROCESS is to select the most beneficial course of project action where alternatives exist.

This process responds to a request for a decision encountered during the system life cycle, whatever its nature or source, in order to reach specified, desirable or optimized outcomes. Alternative actions are analyzed and a course of action selected and directed. Decisions and their rationale are recorded to support future decision-making.

#### 5.3.3.2 OUTCOMES

- a decision-making strategy is defined;
- alternative courses of action are defined;
- a preferred course of action is selected; and
- the resolution, decision rationale and assumptions are captured and reported.

#### 5.3.3.3 ACTIVITIES AND TASKS

##### 1. Decision Planning:

1. The project shall involve relevant parties in the decision-making in order to draw on experience and knowledge.
2. The project shall identify the circumstances and need for a decision.

##### 2. Decision Analysis:

1. The project shall select and declare the decision-making strategy for each decision situation.
2. The project shall identify desired outcomes and measurable success criteria.
3. The project shall evaluate the balance of consequences of alternative actions, using the defined decision-making strategy, to arrive at an optimization of, or an improvement in, an identified decision situation.

##### 3. Decision Tracking:

1. The project shall record, track, evaluate and report decision outcomes to confirm that problems have been effectively resolved, adverse trends have been reversed and advantage has been taken of opportunities.
2. The project shall maintain records of problems and opportunities and their disposition, as stipulated in agreements or organizational procedures and in a manner that permits auditing and learning from experience.

### 5.3.4 RISK MANAGEMENT PROCESS

#### 5.3.4.1 PURPOSE

The purpose of the RISK MANAGEMENT PROCESS is to identify, analyze, treat and monitor the risks continuously.

The Risk Management Process is a continuous process for systematically addressing risk throughout the life cycle of a system or software product or service. It can be applied to risks related to the acquisition, development, maintenance or operation of a system.

#### 5.3.4.2 OUTCOMES

- the scope of risk management to be performed is determined;
- appropriate risk management strategies are defined and implemented;
- risks are identified as they develop and during the conduct of the project;
- risks are analyzed, and the priority in which to apply resources to treatment of these risks is determined;
- risk measures are defined, applied, and assessed to determine changes in the status of risk and the progress of the treatment activities; and
- appropriate treatment is taken to correct or avoid the impact of risk based on its priority, probability, and consequence or other defined risk threshold.

#### 5.3.4.3 ACTIVITIES AND TASKS

##### 1. Risk Management Planning:

1. Risk management policies describing the guidelines under which risk management is to be performed shall be defined.
2. A description of the Risk Management Process to be implemented shall be documented.
3. The parties responsible for performing risk management and their roles and responsibilities shall be identified.
4. The responsible parties shall be provided with adequate resources to perform the Risk Management Process.
5. A description of the process for evaluating and improving the Risk Management Process shall be provided.

##### 2. Risk Profile Management:

1. Risk thresholds, defining the conditions under which a level of risk may be accepted, shall be documented.
2. A risk profile shall be established and maintained.
3. The relevant risk profile shall be communicated periodically to stakeholders based upon their needs.

##### 3. Risk Analysis:

1. Risks shall be identified in the categories described in the risk management context.
2. The probability of occurrence and consequences of each risk identified shall be estimated.
3. Each risk shall be evaluated against its risk thresholds.
4. For each risk that is above its risk threshold, recommended treatment strategies shall be defined and documented. Measures indicating the effectiveness of the treatment alternatives shall also be defined and documented.

5. Stakeholders shall be provided recommended alternatives for risk treatment in risk action requests.
4. **Risk Treatment:**
  1. If the stakeholders determine that actions should be taken to make a risk acceptable, then a risk treatment alternative shall be implemented.
  2. If the stakeholders accept a risk that exceeds its threshold, it shall be considered a high priority and monitored continuously to determine if any future risk treatment actions are necessary.
  3. Once a risk treatment is selected, it shall receive the same management actions as problems do, in accordance with the assessment and control activities in the PROJECT ASSESSMENT AND CONTROL PROCESS.
5. **Risk Monitoring:**
  1. All risks and the risk management context shall be continuously monitored for changes. Risks whose states have changed shall undergo risk evaluation.
  2. Measures shall be implemented and monitored to evaluate the effectiveness of risk treatments.
  3. The project shall continuously monitor for new risks and sources throughout its life cycle.
6. **Risk Management Process Evaluation:**
  1. Information shall be collected throughout the project's life cycle for purposes of improving the Risk Management Process and generating lessons learned.
  2. The Risk Management Process shall be periodically reviewed for its effectiveness and efficiency.
  3. Information on the risks identified, their treatment, and the success of the treatments shall be reviewed periodically for purposes of identifying systemic project and organizational risks.

### 5.3.5 CONFIGURATION MANAGEMENT PROCESS

#### 5.3.5.1 PURPOSE

The purpose of the CONFIGURATION MANAGEMENT PROCESS is to establish and maintain the integrity of all identified outputs of a project or process and make them available to concerned parties.

#### 5.3.5.2 OUTCOMES

- a configuration management strategy is defined;
- items requiring configuration management are defined;
- configuration baselines are established;
- changes to items under configuration management are controlled;
- the configuration of released items is controlled; and
- the status of items under configuration management is made available throughout the life cycle.

#### 5.3.5.3 ACTIVITIES AND TASKS

##### 1. Configuration Management Planning:

1. The project shall define a configuration management strategy.
2. The project shall identify items that are subject to configuration control.

##### 2. Configuration Management Execution:

1. The project shall maintain information on configurations with an appropriate level of integrity and security.
2. The project shall ensure that changes to configuration baselines are properly identified, recorded, evaluated, approved, incorporated and verified.

## 5.3.6 INFORMATION MANAGEMENT PROCESS

### 5.3.6.1 PURPOSE

The purpose of the INFORMATION MANAGEMENT PROCESS is to provide relevant, timely, complete, valid and, if required, confidential information to designated parties during and, as appropriate, after the system life cycle.

This process generates, collects, transforms, retains, retrieves, disseminates and disposes of information. It manages designated information, including technical, project, organizational, agreement and user information.

### 5.3.6.2 OUTCOMES

- information to be managed is identified;
- the forms of the information representations are defined;
- information is transformed and disposed of as required;
- the status of information is recorded;
- information is current, complete and valid; and
- information is made available to designated parties.

### 5.3.6.3 ACTIVITIES AND TASKS

#### 1. Information Management Planning:

1. The project shall define the items of information that will be managed during the system life cycle and, according to organizational policy or legislation, maintained for a defined period beyond.
2. The project shall designate authorities and responsibilities regarding the origination, generation, capture, archiving and disposal of items of information.
3. The project shall define the rights, obligations and commitments regarding the retention of, transmission of and access to information items.
4. The project shall define the content, semantics, formats and medium for the representation, retention, transmission and retrieval of information.
5. The project shall define information maintenance actions.

#### 2. Information Management Execution:

1. The project shall obtain the identified items of information.
2. The project shall maintain information items and their storage records according to integrity, security and privacy requirements.
3. The project shall retrieve and distribute information to designated parties as required by agreed schedules or defined circumstances.
4. The project shall provide official documentation as required.
5. The project shall archive designated information, in accordance with the audit, knowledge retention and project closure purposes.
6. The project shall dispose of unwanted, invalid or unverifiable information according to organization policy, and security and privacy requirements.

## 5.3.7 MEASUREMENT PROCESS

### 5.3.7.1 PURPOSE

The purpose of the MEASUREMENT PROCESS is to collect, analyze, and report data relating to the products developed and processes implemented within the organizational unit, to support effective management of the processes, and to objectively demonstrate the quality of the products.

### 5.3.7.2 OUTCOMES

- the information needs of technical and management processes are identified;
- an appropriate set of measures, driven by the information needs are identified and/or developed;
- measurement activities are identified and planned;
- the required data are collected, stored, analyzed, and the results interpreted;
- information products are used to support decisions and provide an objective basis for communication;
- the Measurement Process and measures are evaluated; and
- improvements are communicated to the Measurement Process owner.

### 5.3.7.3 ACTIVITIES AND TASKS

#### 1. Measurement Planning:

1. The project shall describe the characteristics of the organization that are relevant to measurement.
2. The project shall identify and prioritize the information needs.
3. The project shall select and document measures that satisfy the information needs.
4. The project shall define data collection, analysis, and reporting procedures.
5. The project shall define criteria for evaluating the information products and the measurement process.
6. The project shall review, approve, and provide resources for measurement tasks.
7. The project shall acquire and deploy supporting technologies.

#### 2. Measurement Performance:

1. The project shall integrate procedures for data generation, collection, analysis and reporting into the relevant processes.
2. The project shall collect, store, and verify data.
3. The project shall analyze data and develop information products.
4. The project shall document and communicate results to the measurement users.

#### 3. Measurement Evaluation:

1. The project shall evaluate information products and the measurement process.
2. The project shall identify and communicate potential improvements.

## 5.4 ORGANIZATIONAL PROCESSES

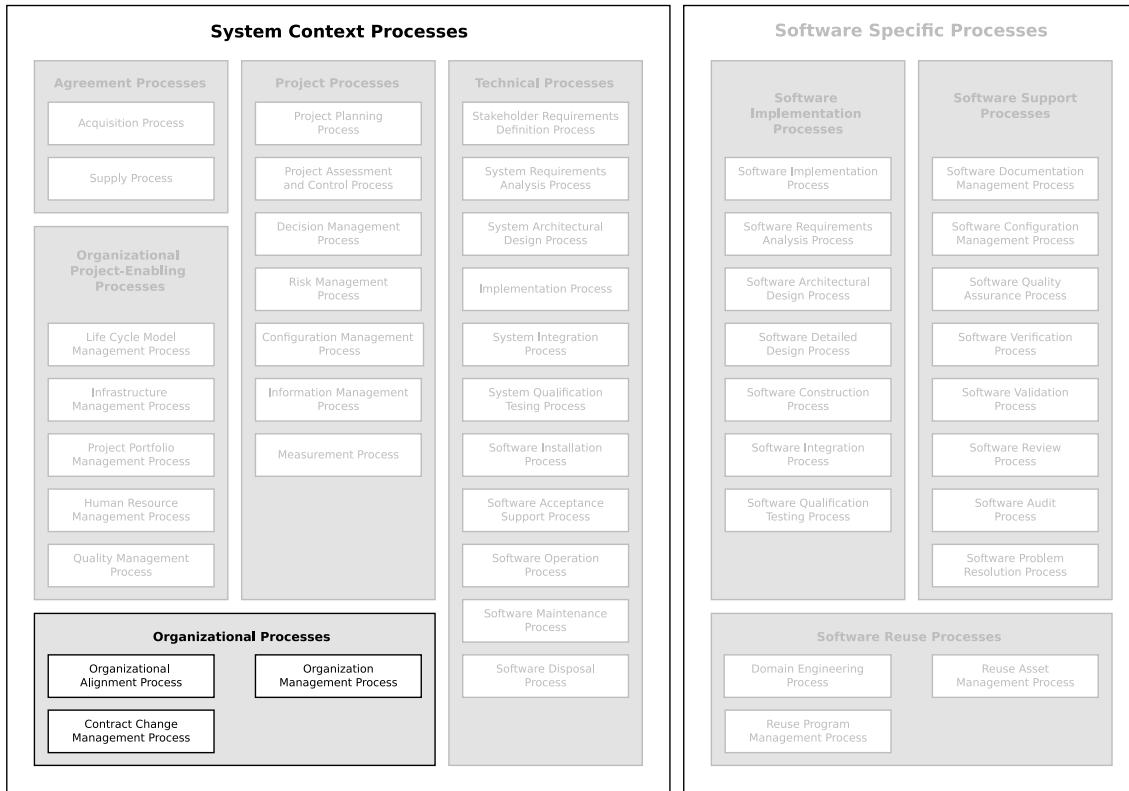


Figure 5: Organizational Processes

The processes within this section are considered highly useful to the management of an organization and its use and management of the processes within this standard.

In this Section:

- 5.4.1 - ORGANIZATIONAL ALIGNMENT PROCESS
- 5.4.2 - ORGANIZATION MANAGEMENT PROCESS
- 5.4.3 - CONTRACT CHANGE MANAGEMENT PROCESS

## 5.4.1 ORGANIZATIONAL ALIGNMENT PROCESS

### 5.4.1.1 PURPOSE

The purpose of the ORGANIZATIONAL ALIGNMENT PROCESS is to enable the software processes needed by the organization to provided software products and services, to be consistent with its goals. Within this context, goals are defined by the business' highest-order business objectives.

### 5.4.1.2 OUTCOMES

- The organization's business goals are identified;
- The process framework is identified and defined that includes a set of software processes needed to achieve the business goals of the organization;
- A strategy is defined for process definition, implementation and improvement;
- Support is provided to enable this strategy;
- The organization's mission, core values, vision, goals and objectives is made known to all employees;
- Individuals in the organization share a common vision, culture, and understanding of the business goals to empower them to function effectively; and
- Everyone in the organization understands their role in achieving the goals of the business and is able to perform that role.

## 5.4.2 ORGANIZATION MANAGEMENT PROCESS

### 5.4.2.1 PURPOSE

The purpose of the ORGANIZATION MANAGEMENT PROCESS is to establish and perform software management practices, during the performance of the processes needed for providing software products and services, that are consistent with the business goals of the organization.

**Note:** Although organizational operations in general have a much broader scope than that of software process, software processes are implemented in a business context and to be effective, require an appropriate organizational environment.

### 5.4.2.2 OUTCOMES

- the organization will invest in the appropriate management infrastructure;
- the best practices are identified to support the implementation of effective organization and project management; and
- a basis for evaluating the achievement of organization business goals based on these management practices is provided.

## 5.4.3 CONTRACT CHANGE MANAGEMENT PROCESS

### 5.4.3.1 PURPOSE

The purpose of the CONTRACT CHANGE MANAGEMENT PROCESS is to develop the new contract contents as agreed by both the acquirer and the supplier when a change request affecting the agreed contract contents is proposed.

This process begins with a proposal of the change request by either the acquirer or the supplier and ends with the conclusion acceptable for both parties: withdrawal or overall/partial approval of the change request.

### 5.4.3.2 OUTCOMES

- the change request to the contract is proposed explicitly and formally;
- the roles and responsibilities of both the acquirer and the supplier for the contract change management are established;
- the impact of the change request to the contract on the project plans, costs, benefits, quality and schedule is evaluated;
- the actions against the change request are taken to get agreement and satisfaction of both the acquirer and the supplier; and
- the result of each change request is made known to all affected parties.

## 5.5 TECHNICAL PROCESSES

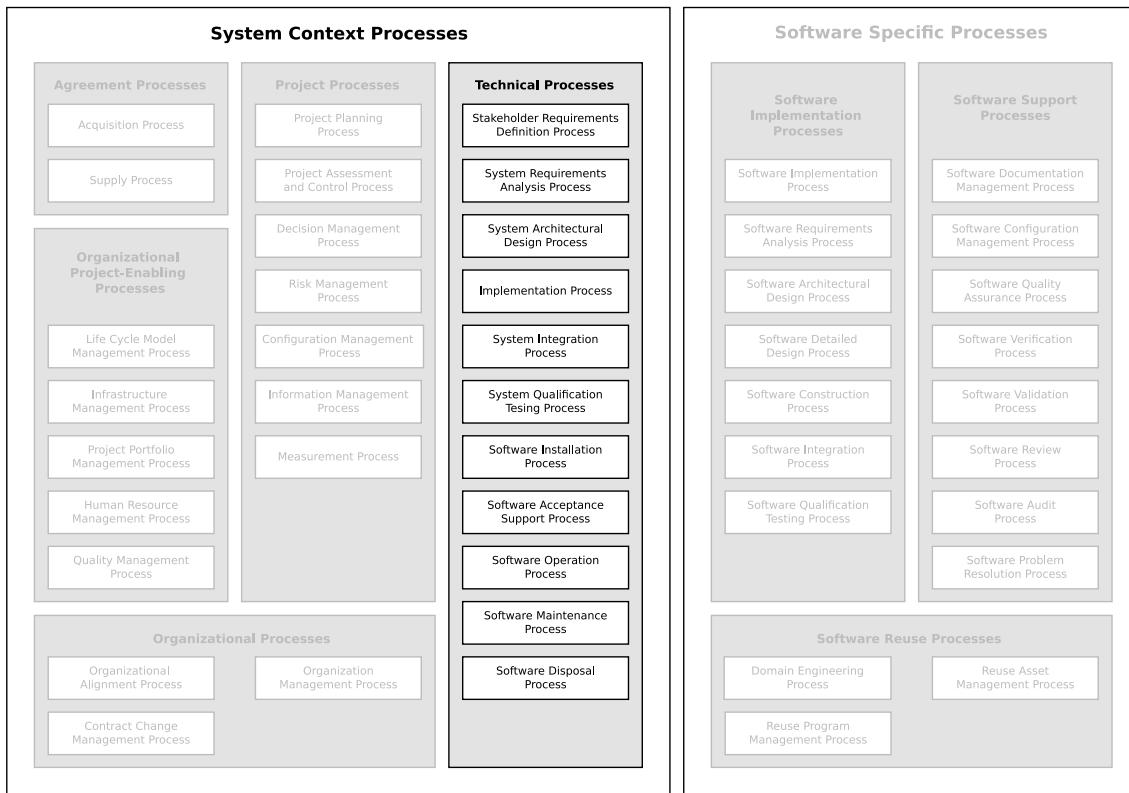


Figure 6: Technical Processes

The Technical Processes define the activities that enable organizational and project functions to optimize the benefits and reduce the risks that arise from technical decisions and actions.

These activities enable products and services to possess the timeliness and availability, the cost effectiveness, and the functionality, reliability, maintainability, producibility, usability and other qualities required by acquiring and supplying organizations. They also enable products and services to conform to the expectations or legislated requirements of society, including health, safety, security and environmental factors.

- 5.5.1 - STAKEHOLDER REQUIREMENTS DEFINITION PROCESS
- 5.5.2 - SYSTEM REQUIREMENTS ANALYSIS PROCESS
- 5.5.3 - SYSTEM ARCHITECTURAL DESIGN PROCESS
- 5.5.4 - IMPLEMENTATION PROCESS
- 5.5.5 - SYSTEM INTEGRATION PROCESS
- 5.5.6 - SYSTEM QUALIFICATION TESTING PROCESS
- 5.5.7 - SOFTWARE INSTALLATION PROCESS
- 5.5.8 - SOFTWARE ACCEPTANCE SUPPORT PROCESS
- 5.5.9 - SOFTWARE OPERATION PROCESS
- 5.5.10 - SOFTWARE MAINTENANCE PROCESS
- 5.5.11 - SOFTWARE DISPOSAL PROCESS

## 5.5.1 STAKEHOLDER REQUIREMENTS DEFINITION PROCESS

### 5.5.1.1 PURPOSE

The purpose of the STAKEHOLDER REQUIREMENTS DEFINITION PROCESS is to define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment.

It identifies stakeholders, or stakeholder classes, involved with the system throughout its life cycle, and their needs and desires. It analyzes and transforms these into a common set of stakeholder requirements that express the intended interaction the system will have with its operational environment and that are the reference against which each resulting operational service is validated in order to confirm that the system fulfills needs.

### 5.5.1.2 OUTCOMES

- the required characteristics and context of use of services are specified;
- the constraints on a system solution are defined;
- traceability of stakeholder requirements to stakeholders and their needs is achieved;
- the basis for defining the system requirements is described;
- the basis for validating the conformance of the services is defined; and
- a basis for negotiating and agreeing to supply a service or product is provided.

### 5.5.1.3 ACTIVITIES AND TASKS

#### 1. Stakeholder Identification:

1. The project shall identify the individual stakeholders or stakeholder classes who have a legitimate interest in the system throughout its life cycle.

#### 2. Requirements Identification:

1. The project shall elicit stakeholder requirements.
2. The project shall define the constraints on a system solution that are unavoidable consequences of existing agreements, management decisions and technical decisions.
3. The project shall define a representative set of activity sequences to identify all required services that correspond to anticipated operational and support scenarios and environments.
4. The project shall identify the interaction between users and the system, taking into the account human capabilities and skills limitations.
  1. Physical, mental, and learned capabilities;
  2. Work place, environment and facilities, including other equipment in the context of use;
  3. Normal, unusual, and emergency conditions;
  4. Operator and user recruitment, training and culture.
5. The project shall specify health, safety, security, environment and other stakeholder requirements and functions that relate to critical qualities and shall address possible adverse effects of use of the system on human health and safety.

#### 3. Requirements Evaluation:

1. The project shall analyze the complete set of elicited requirements.

#### 4. Requirements Agreement:

1. The project shall resolve requirements problems.
  2. The project shall feed back the analyzed requirements to applicable stakeholders to ensure that the needs and expectations have been adequately captured and expressed.
  3. The project shall establish with stakeholders that their requirements are expressed correctly.
5. **Requirements Recording:**
1. The project shall record the stakeholder requirements in a form suitable for requirements management through the life cycle and beyond.
  2. The project shall maintain stakeholder requirements traceability to the sources of stakeholder need.

## 5.5.2 SYSTEM REQUIREMENTS ANALYSIS PROCESS

### 5.5.2.1 PURPOSE

The purpose of SYSTEM REQUIREMENTS ANALYSIS PROCESS is to transform the defined stakeholder requirements into a set of desired system technical requirements that will standard the design of the system.

### 5.5.2.2 OUTCOMES

- a defined set of system functional and non-functional requirements describing the problem to be solved are established;
- the appropriate techniques are performed to optimize the preferred project solution;
- system requirements are analyzed for correctness and testability;
- the impact of the system requirements on the operating environment are understood;
- the requirements are prioritized, approved and updated as needed;
- consistency and traceability are established between the system requirements and the customer's requirements baseline;
- changes to the baseline are evaluated for cost, schedule and technical impact; and
- the system requirements are communicated to all affected parties and baselined.

### 5.5.2.3 ACTIVITIES AND TASKS

#### 1. Requirements Specification:

1. The specific intended use of the system to be developed shall be analyzed to specify system requirements. The system requirements specification shall describe: functions and capabilities of the system; business, organizational and user requirements; safety, security, human-factors engineering (ergonomics), interface, operations, and maintenance requirements; design constraints and qualification requirements. The system requirements specification shall be documented.

#### 2. Requirements Evaluation:

1. The system requirements shall be evaluated considering the criteria listed below. The results of evaluations shall be documented.
  1. Traceability to acquisition needs;
  2. Consistency with acquisition needs;
  3. Testability;
  4. Feasibility of system architectural design;
  5. Feasibility of operation and maintenance.

## 5.5.3 SYSTEM ARCHITECTURAL DESIGN PROCESS

### 5.5.3.1 PURPOSE

The purpose of the SYSTEM ARCHITECTURAL DESIGN PROCESS is to identify which system requirements should be allocated to which elements of the system.

### 5.5.3.2 OUTCOMES

- a system architecture design is defined that identifies the elements of the system and meets the defined requirements;
- the system's functional and non-functional requirements are addressed;
- the requirements are allocated to the elements of the system;
- internal and external interfaces of each system element are defined;
- verification between the system requirements and the system architecture is performed;
- the requirements allocated to the system elements and their interfaces are traceable to the customer's requirements baseline;
- consistency and traceability between the system requirements and system architecture design is maintained; and
- the system requirements, the system architecture design, and their relationships are baselined and communicated to all affected parties;
- human factors and ergonomic knowledge and techniques are incorporated in system design; and
- human-centered design activities are identified and performed.

### 5.5.3.3 ACTIVITIES AND TASKS

#### 1. Establishing Architecture:

1. A top-level architecture of the system shall be established. The architecture shall identify items of hardware, software, and manual operations. It shall be ensured that all the system requirements are allocated among the items. Hardware configuration items, software configuration items, and manual operations shall be subsequently identified from these items. The system architecture and the system requirements allocated to the items shall be documented.

#### 2. Architectural Evaluation:

1. The system architecture and the requirements for the items shall be evaluated considering the criteria listed below. The results of the evaluations shall be documented.
  1. Traceability to the system requirements.
  2. Consistency with the system requirements.
  3. Appropriateness of design standards and methods used.
  4. Feasibility of the software items fulfilling their allocated requirements.
  5. Feasibility of operation and maintenance.

## 5.5.4 IMPLEMENTATION PROCESS

### 5.5.4.1 PURPOSE

The purpose of the IMPLEMENTATION PROCESS is to realize a specified system element.

**Note:** This process is intentionally void of outcomes, activities, and tasks, as the implementation process to realize any given system element is contingent upon the context of that element. See SOFTWARE IMPLEMENTATION PROCESS for how the implementation process is defined for software, as an example.

## 5.5.5 SYSTEM INTEGRATION PROCESS

### 5.5.5.1 PURPOSE

The purpose of the SYSTEM INTEGRATION PROCESS is to integrate the system elements (including software items, hardware items, manual operations, and other systems, as necessary) to produce a complete system that will satisfy the system design and the customers expectations expressed in the system requirements.

### 5.5.5.2 OUTCOMES

- a strategy is developed to integrate the system according to the priorities of the system requirements;
- criteria are developed to verify compliance with the system requirements allocated to the system elements, including the interfaces between system elements;
- the system integration is verified using the defined criteria;
- a regression strategy is developed and applied for re-testing the system when changes are made;
- consistency and traceability are established between the system design and the integrated system elements;
- an integrated system is constructed that demonstrates compliance with the system design; and
- an integrated system is constructed that demonstrates that a complete set of usable deliverable system elements exists.

### 5.5.5.3 ACTIVITIES AND TASKS

#### 1. Integration:

1. The software configuration items shall be integrated, with hardware configuration items, manual operations, and other systems as necessary, into the system. The aggregates shall be tested, as they are developed, against their requirements. The integration and the test results shall be documented.

#### 2. Test Readiness:

1. 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.
2. The integrated system shall be evaluated considering the criteria listed below. The results of the evaluations shall be documented.
  1. Test coverage of system requirements.
  2. Appropriateness of test methods and standards used.
  3. Conformance to expected results.
  4. Feasibility of system qualification testing.
  5. Feasibility of operation and maintenance.

## 5.5.6 SYSTEM QUALIFICATION TESTING PROCESS

### 5.5.6.1 PURPOSE

The purpose of the SYSTEM QUALIFICATION TESTING PROCESS is to ensure that the implementation of each system requirement is tested for compliance and that the system is ready for delivery.

### 5.5.6.2 OUTCOMES

- criteria for evaluating compliance with system requirements are developed;
- the integrated system is tested using the defined criteria;
- test results are recorded; and
- readiness of the system for delivery is assured.

### 5.5.6.3 ACTIVITIES AND TASKS

#### 1. Qualification Testing:

1. System qualification testing shall be conducted in accordance with the qualification requirements specified for the system.
2. It shall be ensured that the implementation of each system requirement is tested for compliance and that the system is ready for delivery.
3. The qualification testing results shall be documented.
4. The system shall be evaluated considering the criteria listed below. evaluations shall be documented.
  1. Test coverage of system requirements;
  2. Conformance to expected results;
  3. Feasibility of operation and maintenance.
5. The developer shall support audit(s) in accordance with SOFTWARE AUDIT PROCESS. The results of the audit(s) shall be documented.
6. Upon successful completion of the audit(s), if conducted, the developer shall update and prepare the deliverable software product for Software Installation and Software Acceptance Support.

## 5.5.7 SOFTWARE INSTALLATION PROCESS

### 5.5.7.1 PURPOSE

The purpose of the SOFTWARE INSTALLATION PROCESS is to install the software product that meets the agreed requirements in the target environment.

### 5.5.7.2 OUTCOMES

- a software installation strategy is developed;
- criteria for software installation are developed that demonstrate compliance with the software installation requirements;
- the software product is installed in the target environment; and
- readiness of the software product for use in its intended environment is assured.

### 5.5.7.3 ACTIVITIES AND TASKS

#### 1. Software Installation:

1. The implementer shall develop a plan to install the software product in the target environment as designated in the contract. The resources and information necessary to install the software product shall be determined and be available. As specified in the contract, the implementer shall assist the acquirer with the set-up activities. Where the installed software product is replacing an existing system, the implementer shall support any parallel running activities that are required by contract. The installation plan shall be documented.
2. The developer shall install the software product in accordance with the installation plan. It shall be ensured that the software code and databases initialize, execute, and terminate as specified in the contract. The installation events and results shall be documented.

## 5.5.8 SOFTWARE ACCEPTANCE SUPPORT PROCESS

### 5.5.8.1 PURPOSE

The purpose of the SOFTWARE ACCEPTANCE SUPPORT PROCESS is to assist the acquirer to achieve confidence that the product meets requirements.

### 5.5.8.2 OUTCOMES

- the product is completed and delivered to the acquirer;
- acquirer acceptance tests and reviews are supported;
- the product is put into operation in the customers environment; and
- problems detected during acceptance are identified and communicated to those responsible for resolution.

### 5.5.8.3 ACTIVITIES AND TASKS

#### 1. Software Acceptance Support:

1. The developer shall support the acquirer's acceptance review and testing of the software product. Acceptance review and testing shall consider the results of the SOFTWARE REVIEW PROCESS, SOFTWARE AUDIT PROCESS, SOFTWARE QUALIFICATION TESTING PROCESS, and SYSTEM QUALIFICATION TESTING PROCESS (if performed) processes. The results of the acceptance review and testing shall be documented.
2. The developer shall complete and deliver the software product as specified in the contract.
3. The developer shall provide initial and continuing training and support to the acquirer as specified in the contract.

## 5.5.9 SOFTWARE OPERATION PROCESS

### 5.5.9.1 PURPOSE

The purpose of the SOFTWARE OPERATION PROCESS is to operate the software product in its intended environment and to provide support to the customers of the software product.

### 5.5.9.2 OUTCOMES

- an operation strategy is defined;
- conditions for correct operation of the software in its intended environment are identified and evaluated;
- the software is tested and determined to operate in its intended environment;
- the software is operated in its intended environment; and
- assistance and consultation is provided to the customers of the software product in accordance with the agreement.

### 5.5.9.3 ACTIVITIES AND TASKS

#### 1. Preparation for Operation:

1. The operator shall develop a plan and set operational standards for performing the activities and tasks of this process. The plan shall be documented and executed.
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 SOFTWARE PROBLEM RESOLUTION PROCESS.
3. The operator shall establish procedures for testing the software product in its operation environment, for entering problem reports and modification requests to the SOFTWARE MAINTENANCE PROCESS, and for releasing the software product for operational use.

#### 2. Operation Activation and Check-out:

1. For each release of the software product, the operator shall perform operational testing, and, on satisfying the specified criteria, release the software product for operational use.
2. The operator shall ensure that the software code and databases initialize, execute, and terminate as described in the plan.
3. The operator shall activate the system in its intended operational situation to deliver instances of service or continuous service according to its intended purpose.

#### 3. Operational Use:

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

#### 4. Customer Support:

1. The operator shall provide assistance and consultation to the users as requested. These requests and subsequent actions shall be recorded and monitored.
2. The operator shall forward user requests, as necessary, to the SOFTWARE 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. Operational Problem Resolution:

1. The operator shall forward identified problems to the Software Problem Resolution Process for

resolution.

2. 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 SOFTWARE MAINTENANCE PROCESS.

## 5.5.10 SOFTWARE MAINTENANCE PROCESS

### 5.5.10.1 PURPOSE

The purpose of the SOFTWARE MAINTENANCE PROCESS is to provide cost-effective support to a delivered software product.

**Note:** Pre-delivery Software Maintenance activities include planning for post-delivery operations, supportability, and logistics determination. Post-delivery activities include software modification and operational support, such as training or operating a help desk.

### 5.5.10.2 OUTCOMES

- a maintenance strategy is developed to manage modification and migration of products according to the release strategy;
- the impact of changes to the existing system on organization, operations or interfaces are identified;
- affected system and software documentation is updated as needed;
- modified products are developed with associated tests that demonstrate that requirements are not compromised;
- product upgrades are migrated to the customers environment; and
- the system software modification is communicated to all affected parties.

### 5.5.10.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. The maintainer shall develop, document, and execute plans and procedures for conducting the activities and tasks of the Software Maintenance Process.
2. The maintainer shall establish procedures for receiving, recording, and tracking problem reports and modification requests from the users and providing feedback to the users. Whenever problems are encountered, they shall be recorded and entered into the SOFTWARE PROBLEM RESOLUTION PROCESS.
3. The maintainer shall implement (or establish organizational interface with) the CONFIGURATION MANAGEMENT PROCESS for managing modifications to the existing system.

#### 2. Problem and Modification Analysis:

1. The maintainer shall analyze the problem report or modification request for its impact on the organization, the existing system, and the interfacing systems for the following:
  1. Type; for example, corrective, improvement, preventive, or adaptive to new environment;
  2. Scope; for example, size of modification, cost involved, time to modify;
  3. Criticality; for example, impact on performance, safety, or security.
2. The maintainer shall replicate or verify the problem.
3. The maintainer shall document the problem/modification request, the analysis results, and implementation options.
4. The maintainer shall obtain approval for the selected modification option as specified in the contract.

#### 3. Modification Implementation:

1. The maintainer shall conduct analysis and determine which documentation, software units, and

- versions thereof need to be modified. These shall be documented.
2. The maintainer shall enter the TECHNICAL PROCESSES to implement the modifications. The requirements of the Technical Processes shall be supplemented as follows:
    3. 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.
    4. 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.
  4. **Maintenance Review/Acceptance:**
    1. The maintainer shall conduct review(s) with the organization authorizing the modification to determine the integrity of the modified system.
    2. The maintainer shall obtain approval for the satisfactory completion of the modification as specified in the contract.
  5. **Migration:**
    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 is in accordance with this standard.
    2. A migration plan shall be developed, documented, and executed. The planning activities shall include users. Items included in the plan shall include the following:
      1. Requirements analysis and definition of migration.
      2. Development of migration tools.
      3. Conversion of software product and data.
      4. Migration execution.
      5. Migration verification.
      6. Support for the old environment in the future.
    3. Users shall be given notification of the migration plans and activities. Notifications shall include the following:
      1. Statement of why the old environment is no longer to be supported.
      2. Description of the new environment with its date of availability.
      3. Description of other support options available, if any, once support for the old environment has been removed.
    4. Parallel operations of the old and new environments may be conducted for smooth transition to the new environment. During this period, necessary training shall be provided as specified in the contract.
    5. When the scheduled migration arrives, notification shall be sent to all concerned. Associated old environment documentation, logs, and code should be placed in archives.
    6. A post-operation review shall be performed to assess the impact of changing to the new environment. The results of the review shall be sent to the appropriate authorities for information, guidance, and action.
    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.

## 5.5.11 SOFTWARE DISPOSAL PROCESS

### 5.5.11.1 PURPOSE

The purpose of the SOFTWARE DISPOSAL PROCESS is to end the existence of a system's software entity. This process ends active support by the operation and maintenance organization, or deactivates, disassembles and removes the affected software products, consigning them to a final condition and leaving the environment in an acceptable condition.

This process destroys or stores system software elements and related products in a sound manner, in accordance with legislation, agreements, organizational constraints and stakeholder requirements. Where required, it maintains records that may be monitored.

**Note:** The objective is to retire a system's existing software products or services while preserving the integrity of organizational operations.

### 5.5.11.2 OUTCOMES

- a software disposal strategy is defined;
- disposal constraints are provided as inputs to requirements;
- the system's software elements are destroyed or stored;
- the environment is left in an agreed-upon state; and
- records allowing knowledge retention of disposal actions and any analysis of long-term impacts are available.

### 5.5.11.3 ACTIVITIES AND TASKS

#### 1. Software Disposal Planning:

1. A software disposal strategy is defined and documented. A plan to remove active support by the operation and maintenance organizations shall be developed and documented. The planning activities shall include users. The software disposal plan shall address the items listed below:
  1. Cessation of full or partial support after a certain period of time.
  2. Archiving of the software product and its associated documentation.
  3. Responsibility for any future residual support issues.
  4. Transition to any new software product, if applicable.
  5. Accessibility of archive copies of data.

#### 2. Software Disposal Execution:

1. The software disposal plan shall be executed.
2. Users shall be given notification of the plans and activities for the retirement of software products and services. Notifications shall include the following:
  1. Description of any replacement or upgrade with its date of availability.
  2. Statement of why the software product is no longer to be supported.
  3. Description of other support options available, once support has been removed.
3. Parallel operations of the retiring and any 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.

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

## 6. SOFTWARE SPECIFIC PROCESSES

### 6.1 SOFTWARE IMPLEMENTATION PROCESSES

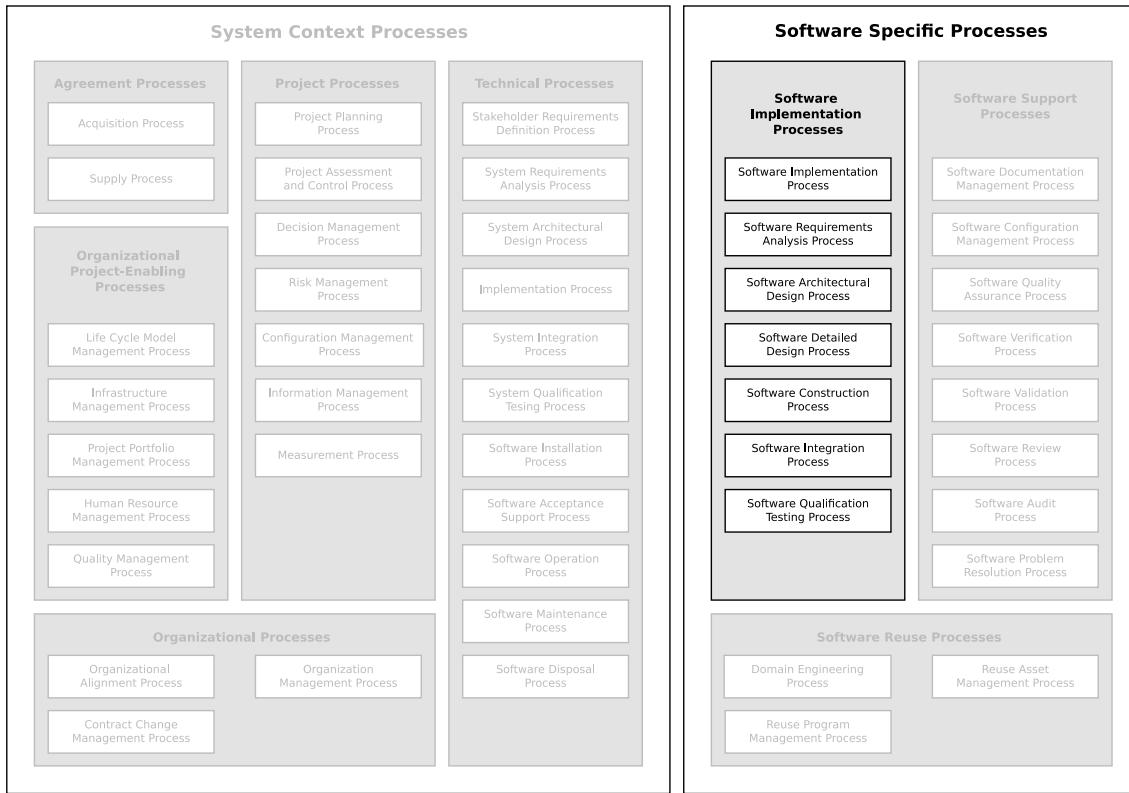


Figure 7: Software Implementation Processes

The SOFTWARE IMPLEMENTATION PROCESSES are used to produce a specified system element (software item) implemented in software. Those processes transform specified behavior, interfaces and implementation constraints into implementation actions resulting in a system element that satisfies the requirements derived from the system requirements.

- 6.1.1 - SOFTWARE IMPLEMENTATION PROCESS
- 6.1.2 - SOFTWARE REQUIREMENTS ANALYSIS PROCESS
- 6.1.3 - SOFTWARE ARCHITECTURAL DESIGN PROCESS
- 6.1.4 - SOFTWARE DETAILED DESIGN PROCESS
- 6.1.5 - SOFTWARE DETAILED DESIGN PROCESS
- 6.1.5 - SOFTWARE CONSTRUCTION PROCESS
- 6.1.6 - SOFTWARE INTEGRATION PROCESS
- 6.1.7 - SOFTWARE QUALIFICATION TESTING PROCESS

## 6.1.1 SOFTWARE IMPLEMENTATION PROCESS

### 6.1.1.1 PURPOSE

The purpose of the SOFTWARE IMPLEMENTATION PROCESS is to produce a specified system element implemented as a software product or service.

This process transforms specified behavior, interfaces and implementation constraints into actions that create a system element implemented as a software product or service, otherwise known as a “software item”.

This process results in a software item that satisfies architectural design requirements through verification and stakeholder requirements through validation.

### 6.1.1.2 OUTCOMES

- an implementation strategy is defined;
- implementation technology constraints on the design are identified;
- a software item is realized; and
- a software item is packaged and stored in accordance with an agreement for its supply.
- In addition to its activities, this process has the following lower-level processes:
  1. Software Architectural Design Process
  2. Software Detailed Design Process
  3. Software Construction Process
  4. Software Integration Process
  5. Software Qualification Testing Process

### 6.1.1.3 ACTIVITIES AND TASKS

#### 1. Software Implementation Strategy:

1. If not stipulated in the contract, the developer shall define or select a life cycle model appropriate to the scope, magnitude, and complexity of the project. The life cycle model shall be comprised of stages and the purpose and outcomes of each stage. The activities and tasks of the Software Implementation Process shall be selected and mapped onto the life cycle model.
2. The implementer shall:
  1. Document the outputs in accordance with the SOFTWARE DOCUMENTATION MANAGEMENT PROCESS.
  2. Place the outputs under the SOFTWARE CONFIGURATION MANAGEMENT PROCESS and perform change control in accordance with it.
  3. Document and resolve problems and non-conformances found in the software products and tasks in accordance with the SOFTWARE PROBLEM RESOLUTION PROCESS.
  4. Perform supporting processes as specified in the contract.
  5. Establish baselines and incorporate configuration items at appropriate times, as determined by the acquirer and the supplier.
3. The implementer 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 Software Implementation Process and supporting processes.

4. The implementer shall develop plans for conducting the activities of the Software Implementation process. The plans should include specific standards, methods, tools, actions, and responsibility associated with the development and qualification of all requirements including safety and security. If necessary, separate plans may be developed. These plans shall be documented and executed.
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.

## 6.1.2 SOFTWARE REQUIREMENTS ANALYSIS PROCESS

### 6.1.2.1 PURPOSE

The purpose of SOFTWARE REQUIREMENTS ANALYSIS PROCESS is to establish the requirements of the software elements of the system.

### 6.1.2.2 OUTCOMES

- the requirements allocated to the software elements of the system and their interfaces are defined;
- software requirements are analyzed for correctness and testability;
- the impact of software requirements on the operating environment are understood;
- consistency and traceability are established between the software requirements and system requirements;
- prioritization for implementing the software requirements is defined;
- the software requirements are approved and updated as needed;
- changes to the software requirements are evaluated for cost, schedule and technical impact; and
- the software requirements are baselined and communicated to all affected parties.

### 6.1.2.3 ACTIVITIES AND TASKS

#### 1. Software Requirements Analysis:

1. The implementer shall establish and document software requirements (including the quality characteristics specifications) described below.
  1. Functional and capability specifications, including performance, physical characteristics, environmental conditions under which the software item is to perform.
  2. Interfaces external to the software item.
  3. Qualification requirements.
  4. Safety specifications, including those related to methods of operation and maintenance, environmental influences, and personnel injury.
  5. Security specifications, including those related to compromise of sensitive information.
  6. Human-factors engineering (ergonomics) specifications, 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.
  7. Data definition and database requirements.
  8. Installation and acceptance requirements of the delivered software product at the operation and maintenance site(s).
  9. User documentation requirements.
  10. User operation and execution requirements.
  11. User maintenance requirements.
2. The implementer shall evaluate the software requirements considering the criteria listed below. The results of the evaluations shall be documented.
  1. Traceability to system requirements and system design.
  2. External consistency with system requirements.
  3. Internal consistency.
  4. Testability.

- 
- 5. Feasibility of software design.
  - 6. Feasibility of operation and maintenance.
  - 3. The implementer shall conduct review(s) in accordance with SOFTWARE REVIEW PROCESS.

### 6.1.3 SOFTWARE ARCHITECTURAL DESIGN PROCESS

#### 6.1.3.1 PURPOSE

The purpose of the SOFTWARE ARCHITECTURAL DESIGN PROCESS is to provide a design for the software that implements and can be verified against the requirements.

#### 6.1.3.2 OUTCOMES

- a software architectural design is developed and baselined that describes the software items that will implement the software requirements;
- internal and external interfaces of each software item are defined; and
- consistency and traceability are established between software requirements and software design.

#### 6.1.3.3 ACTIVITIES AND TASKS

##### 1. Software Architectural Design:

1. The implementer shall transform the requirements for the software item into an architecture that describes its top-level structure and identifies the software components. It shall be ensured that all the requirements for the software item are allocated to its software components and further refined to facilitate detailed design. The architecture of the software item shall be documented.
2. The implementer shall develop and document a top-level design for the interfaces external to the software item and between the software components of the software item.
3. The implementer shall develop and document a top-level design for the database.
4. The implementer should develop and document preliminary versions of user documentation.
5. The implementer shall define and document preliminary test requirements and the schedule for Software Integration.
6. The implementer shall evaluate the architecture of the software item and the interface and database designs considering the criteria listed below. The results of the evaluations shall be documented.
  1. Traceability to the requirements of the software item.
  2. External consistency with the requirements of the software item.
  3. Internal consistency between the software components.
  4. Appropriateness of design methods and standards used.
  5. Feasibility of detailed design.
  6. Feasibility of operation and maintenance.
7. The implementer shall conduct review(s) in accordance with SOFTWARE REVIEW PROCESS

## 6.1.4 SOFTWARE DETAILED DESIGN PROCESS

### 6.1.4.1 PURPOSE

The purpose of the SOFTWARE DETAILED DESIGN PROCESS is to provide a design for the software that implements and can be verified against the requirements and the software architecture and is sufficiently detailed to permit coding and testing.

### 6.1.4.2 OUTCOMES

- a detailed design of each software component, describing the software units to be built, is developed;
- external interfaces of each software unit are defined; and
- consistency and traceability are established between the detailed design and the requirements and architectural design.

### 6.1.4.3 ACTIVITIES AND TASKS

#### 1. Software Detailed Design:

1. The implementer 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. The detailed design shall be documented.
2. The implementer shall develop and document a detailed design for the interfaces external to the software item, between the software components, and between the software units. The detailed design of the interfaces shall permit coding without the need for further information.
3. The implementer shall develop and document a detailed design for the database.
4. The implementer shall update user documentation as necessary.
5. The implementer shall define and document test requirements and the schedule for testing software units. The test requirements should include stressing the software unit at the limits of its requirements.
6. The implementer shall update the test requirements and the schedule for Software Integration.
7. The implementer shall evaluate the software detailed design and test requirements considering the criteria listed below. The results of the evaluations shall be documented.
  1. Traceability to the requirements of the software item;
  2. External consistency with architectural design;
  3. Internal consistency between software components and software units;
  4. Appropriateness of design methods and standards used;
  5. Feasibility of testing;
  6. Feasibility of operation and maintenance.
8. The implementer shall conduct review(s) in accordance with the SOFTWARE REVIEW PROCESS.

## 6.1.5 SOFTWARE CONSTRUCTION PROCESS

### 6.1.5.1 PURPOSE

The purpose of the SOFTWARE CONSTRUCTION PROCESS is to produce executable software units that properly reflect the software design.

### 6.1.5.2 OUTCOMES

- verification criteria are defined for all software units against their requirements;
- software units defined by the design are produced;
- consistency and traceability are established between software units and requirements and design; and
- verification of the software units against the requirements and the design is accomplished.

### 6.1.5.3 ACTIVITIES AND TASKS

#### 1. Software Construction:

1. The implementer shall develop and document the following:
  2. Each software unit and database.
  3. Test procedures and data for testing each software unit and database.
  4. The implementer shall test each software unit and database ensuring that it satisfies its requirements. The test results shall be documented.
  5. The implementer shall update the user documentation as necessary.
  6. The implementer shall update the test requirements and the schedule for Software Integration.
  7. The implementer shall evaluate software code and test results considering the criteria listed below. The results of the evaluations shall be documented.
    1. Traceability to the requirements and design of the software item.
    2. External consistency with the requirements and design of the software item.
    3. Internal consistency between unit requirements.
    4. Test coverage of units.
    5. Appropriateness of coding methods and standards used.
    6. Feasibility of software integration and testing.
    7. Feasibility of operation and maintenance.

## 6.1.6 SOFTWARE INTEGRATION PROCESS

### 6.1.6.1 PURPOSE

The purpose of the SOFTWARE INTEGRATION PROCESS is to combine the software units and software components, producing integrated software items, consistent with the software design, that demonstrate that the functional and non-functional software requirements are satisfied on an equivalent or complete operational platform.

### 6.1.6.2 OUTCOMES

- an integration strategy is developed for software units consistent with the software design and the prioritized software requirements;
- verification criteria for software items are developed that ensure compliance with the software requirements allocated to the items;
- software items are verified using the defined criteria;
- software items defined by the integration strategy are produced;
- results of integration testing are recorded;
- consistency and traceability are established between software design and software items; and
- a regression strategy is developed and applied for re-verifying software items when a change in software units (including associated requirements, design and code) occur.

### 6.1.6.3 ACTIVITIES AND TASKS

#### 1. Software Integration:

1. The implementer shall develop an integration plan to integrate the software units and software components into the software item. The plan shall include test requirements, procedures, data, responsibilities, and schedule. The plan shall be documented.
2. The implementer shall integrate the software units and software components and test as the aggregates are developed in accordance with the integration plan. It shall be ensured that each aggregate satisfies the requirements of the software item and that the software item is integrated at the conclusion of the integration activity. The integration and test results shall be documented.
3. The implementer shall update the user documentation as necessary.
4. The implementer 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. The implementer shall evaluate the integration plan, design, code, tests, test results, and user documentation considering the criteria listed below. The results of the evaluations shall be documented.
  1. Traceability to the system requirements.
  2. External consistency with the system requirements.
  3. Internal consistency.
  4. Test coverage of the requirements of the software item.
  5. Appropriateness of test standards and methods used.
  6. Conformance to expected results.

- 
- 7. Feasibility of software qualification testing.
  - 8. Feasibility of operation and maintenance.

## 6.1.7 SOFTWARE QUALIFICATION TESTING PROCESS

### 6.1.7.1 PURPOSE

The purpose of the SOFTWARE QUALIFICATION TESTING PROCESS is to confirm that the integrated software product meets its defined requirements.

### 6.1.7.2 OUTCOMES

- criteria for the integrated software is developed that demonstrates compliance with the software requirements;
- integrated software is verified using the defined criteria;
- test results are recorded; and
- a regression strategy is developed and applied for re-testing the integrated software when a change in software items is made.

### 6.1.7.3 ACTIVITIES AND TASKS

#### 1. Software Qualification Testing:

1. The implementer 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.
2. The implementer shall update the user documentation as necessary.
3. The implementer shall evaluate the design, code, tests, test results, and user documentation considering the criteria listed below. The results of the evaluations shall be documented.
  1. Test coverage of the requirements of the software item.
  2. Conformance to expected results.
  3. Feasibility of system integration and testing, if conducted.
  4. Feasibility of operation and maintenance.
4. The implementer shall support audit(s) in accordance with the SOFTWARE AUDIT PROCESS. The results of the audits shall be documented. If both hardware and software are under development or integration, the audits may be postponed until the System Qualification Testing.
5. Upon successful completion of the audits, if conducted, the implementer shall update and prepare the deliverable software product for the SYSTEM INTEGRATION PROCESS, SYSTEM QUALIFICATION TESTING PROCESS, SOFTWARE INSTALLATION PROCESS, or SOFTWARE ACCEPTANCE SUPPORT PROCESS accordingly.

## 6.2 SOFTWARE SUPPORT PROCESSES

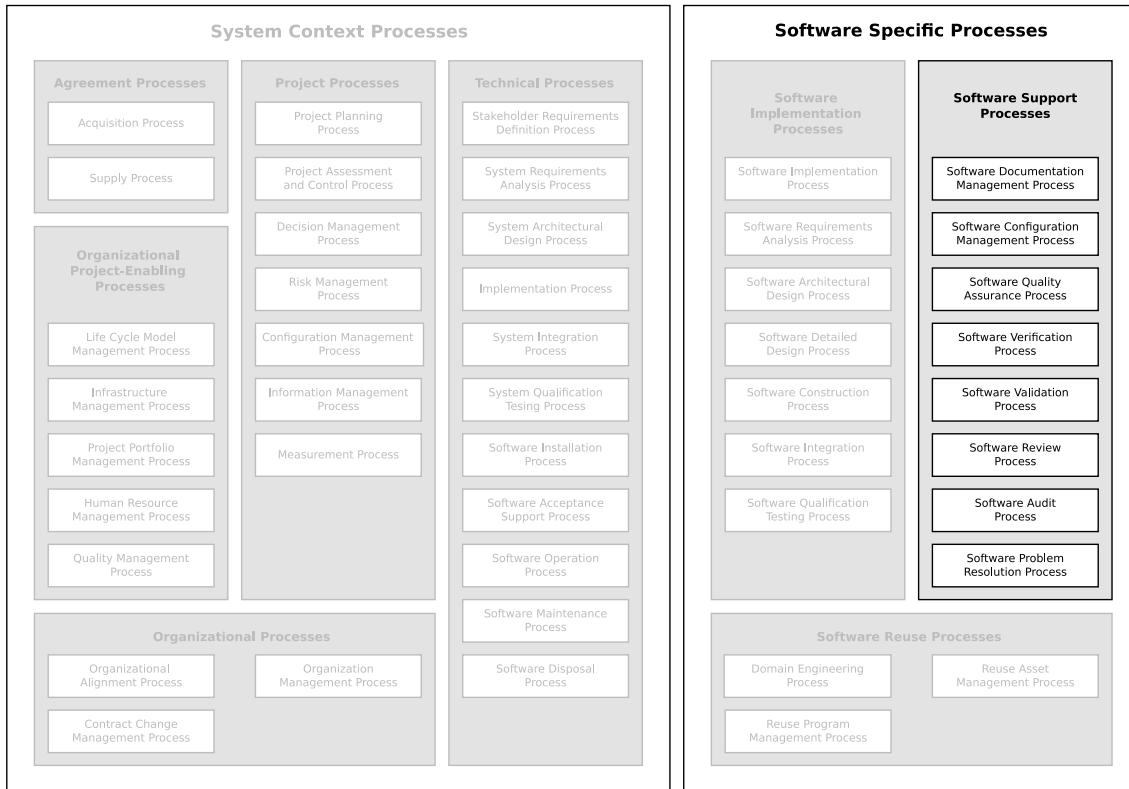


Figure 8: Software Support Processes

**Note:** The support processes listed in this section are specific to software and are labeled SOFTWARE SUPPORT PROCESSES. Although they play an integral role in assisting the SOFTWARE IMPLEMENTATION PROCESS, the Software Support Processes may also provide services to other processes, e.g., the AGREEMENT PROCESSES, SYSTEM QUALIFICATION TESTING PROCESS, SOFTWARE ACCEPTANCE SUPPORT PROCESS, SOFTWARE OPERATION PROCESS, and SOFTWARE MAINTENANCE PROCESS.

- 6.2.1 - SOFTWARE DOCUMENTATION MANAGEMENT PROCESS
- 6.2.2 - SOFTWARE CONFIGURATION MANAGEMENT PROCESS
- 6.2.3 - SOFTWARE QUALITY ASSURANCE PROCESS
- 6.2.4 - SOFTWARE VERIFICATION PROCESS
- 6.2.5 - SOFTWARE VALIDATION PROCESS
- 6.2.6 - SOFTWARE REVIEW PROCESS
- 6.2.7 - SOFTWARE AUDIT PROCESS
- 6.2.8 - SOFTWARE PROBLEM RESOLUTION PROCESS

## 6.2.1 SOFTWARE DOCUMENTATION MANAGEMENT PROCESS

### 6.2.1.1 PURPOSE

The purpose of the SOFTWARE DOCUMENTATION MANAGEMENT PROCESS is to develop and maintain the recorded software information produced by a process.

### 6.2.1.2 OUTCOMES

- a strategy identifying the documentation to be produced during the life cycle of the software product or service is developed;
- the standards to be applied for the development of the software documentation are identified;
- documentation to be produced by the process or project is identified;
- the content and purpose of all documentation is specified, reviewed and approved;
- documentation is developed and made available in accordance with identified standards; and
- documentation is maintained in accordance with defined criteria.

### 6.2.1.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. A plan, identifying the documents to be produced during the life cycle of the software product shall be developed, documented, and implemented. For identified documentation, the following shall be addressed:
  1. Title or name.
  2. Purpose and content.
  3. Intended audience.
  4. Procedures and responsibilities for inputs, development, review, modification, approval, production, storage, distribution, maintenance, and configuration management.
  5. Schedule for intermediate and final versions.

#### 2. Design and Development:

1. Each identified document shall be designed in accordance with applicable documentation standards for medium, format, content description, page numbering, figure/table placement, proprietary/security marking, packaging, and other presentation items.
2. The source and appropriateness of input data for the documents shall be confirmed. Automated documentation support tools may be used.
3. The prepared documents shall be reviewed and edited for format, technical content, and presentation style against their documentation standards. They shall be approved for adequacy by authorized personnel prior to issue.

#### 3. Production:

1. The documents shall be produced and provided in accordance with the plan. Production and distribution of documents may use paper, electronic, or other media. Master materials shall be stored in accordance with requirements for record retention, security, maintenance, and backup.
2. Controls shall be established in accordance with the SOFTWARE CONFIGURATION MANAGEMENT PROCESS.

#### 4. Maintenance:

1. The tasks of the SOFTWARE MAINTENANCE PROCESS, that are required when documentation is to be modified, shall be performed. For those documents that are under configuration management, modifications shall be managed in accordance with the SOFTWARE CONFIGURATION MANAGEMENT PROCESS.

## 6.2.2 SOFTWARE CONFIGURATION MANAGEMENT PROCESS

### 6.2.2.1 PURPOSE

The purpose of the SOFTWARE CONFIGURATION MANAGEMENT PROCESS is to establish and maintain the integrity of the software items of a process or project and make them available to concerned parties.

### 6.2.2.2 OUTCOMES

- a software configuration management strategy is developed;
- items generated by the process or project are identified, defined and baselined;
- modifications and releases of the items are controlled;
- modifications and releases are made available to affected parties;
- the status of the items and modifications are recorded and reported;
- the completeness and consistency of the items is ensured; and
- the storage, handling and delivery of the items are controlled.

### 6.2.2.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. A software configuration management plan shall be developed. The plan shall describe: the configuration management activities; procedures and schedule for performing these activities; the organization(s) responsible for performing these activities; and their relationship with other organizations, such as software development or maintenance. The plan shall be documented and implemented.

#### 2. Configuration Identification:

1. A scheme shall be established for identification of software items and their versions to be controlled for the project. For each software item and its versions, the following shall be identified: the documentation that establishes the baseline; the version references; and other identification details.

#### 3. Configuration Control:

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.

#### 4. Configuration Status Accounting:

1. Management records and status reports that show the status and history of controlled software items, including baselines shall be prepared. Status reports should include the number of changes for a project, latest software item versions, release identifiers, the number of releases, and comparisons of releases.

#### 5. Configuration Evaluation:

1. The following shall be determined and ensured: the functional completeness of the software items against their requirements and the physical completeness of the software items (whether their design and code reflect an up-to-date technical description).

#### 6. Release Management and Delivery:

1. The release and delivery of software products and documentation shall be formally controlled. Master copies of code and documentation shall be maintained for the life of the software product. The code and documentation that contain safety or security critical functions shall be handled, stored, packaged, and delivered in accordance with the policies of the organizations involved.

### 6.2.3 SOFTWARE QUALITY ASSURANCE PROCESS

#### 6.2.3.1 PURPOSE

The purpose of the SOFTWARE QUALITY ASSURANCE PROCESS is to provide assurance that work products and processes comply with predefined provisions and plans.

#### 6.2.3.2 OUTCOMES

- a strategy for conducting quality assurance is developed;
- evidence of software quality assurance is produced and maintained;
- problems and/or non-conformance with requirements are identified and recorded; and
- adherence of products, processes and activities to the applicable standards, procedures and requirements are verified.

#### 6.2.3.3 ACTIVITIES AND TASKS

##### 1. Process Implementation:

1. A quality assurance process suited 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.
2. The quality assurance process should be coordinated with the related SOFTWARE VERIFICATION PROCESS, SOFTWARE VALIDATION PROCESS, SOFTWARE REVIEW PROCESS, and SOFTWARE AUDIT PROCESS.
3. A plan for conducting the quality assurance process activities and tasks shall be developed, documented, implemented, and maintained for the life of the contract. The plan shall include the following:
  1. Quality standards, methodologies, procedures, and tools for performing the quality assurance activities (or their references in organization's official documentation).
  2. Procedures for contract review and coordination thereof.
  3. Procedures for identification, collection, filing, maintenance, and disposition of quality records.
  4. Resources, schedule, and responsibilities for conducting the quality assurance activities.
  5. Selected activities and tasks from supporting processes, such as the SOFTWARE VERIFICATION PROCESS, SOFTWARE VALIDATION PROCESS, SOFTWARE REVIEW PROCESS, SOFTWARE AUDIT PROCESS, and SOFTWARE PROBLEM RESOLUTION PROCESS.
4. Scheduled and on-going quality assurance activities and tasks shall be executed. When problems or non-conformances with contract requirements are detected, they shall be documented and serve as input to the SOFTWARE PROBLEM RESOLUTION PROCESS. Records of these activities and tasks, their execution, problems, and problem resolutions shall be prepared and maintained.
5. Records of quality assurance activities and tasks shall be made available to the acquirer as specified in the contract.
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.

**2. Product Assurance:**

1. It shall be assured that all the plans required by the contract are documented, comply with the contract, are mutually consistent, and are being executed as required.
2. It shall be assured that software products and related documentation comply with the contract and adhere to the plans.
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.

**3. Process Assurance:**

1. It shall be assured that those software life cycle processes (supply, development, operation, maintenance, and support processes including quality assurance) employed for the project comply with the contract and adhere to the plans.
2. It shall be assured that the internal software engineering practices, development environment, test environment, and libraries comply with the contract.
3. It shall be assured that applicable prime-contract requirements are passed down to the subcontractor, and that the subcontractor's software products satisfy prime-contract requirements.
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.
5. It should be assured that software product and process measurements are in accordance with established standards and procedures.
6. It shall be assured that the staff assigned have the skill and knowledge needed to meet the requirements of the project and receive any necessary training.

## 6.2.4 SOFTWARE VERIFICATION PROCESS

### 6.2.4.1 PURPOSE

The purpose of the SOFTWARE VERIFICATION PROCESS is to confirm that each software work product and/or service of a process or project properly reflects the specified requirements.

### 6.2.4.2 OUTCOMES

- a verification strategy is developed and implemented;
- criteria for verification of all required software work products is identified;
- required verification activities are performed;
- defects are identified and recorded; and
- results of the verification activities are made available to the customer and other involved parties.

### 6.2.4.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

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:
  1. 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.
  2. The maturity of and risks associated with the software technology to be used.
  3. Availability of funds and resources.
2. If the project warrants a verification effort, a verification process shall be established to verify the software product.
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.
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 the SOFTWARE VERIFICATION PROCESS, including associated methods, techniques, and tools for performing the tasks, shall be selected for the target life cycle activities and software products.
5. Based upon the verification tasks as determined, a verification plan shall be developed and documented. The plan shall address the life cycle activities and software products subject to verification, the required verification tasks for each life cycle activity and software product, and related resources, responsibilities, and schedule. The plan shall address procedures for forwarding verification reports to the acquirer and other involved organizations.
6. The verification plan shall be implemented. Problems and non-conformances detected by the verification effort shall be entered into the SOFTWARE PROBLEM RESOLUTION PROCESS. All problems and non-conformances shall be resolved. Results of the verification activities shall be made available to the acquirer and other involved organizations.

#### 2. Requirements Verification:

1. The system requirements are consistent, feasible, and testable.
  2. The system requirements have been appropriately allocated to hardware items, software items, and manual operations according to design criteria.
  3. The software requirements are consistent, feasible, testable, and accurately reflect system requirements.
  4. The software requirements related to safety, security, and criticality are correct as shown by suitably rigorous methods.
- 3. Design Verification:**
1. The design is correct and consistent with and traceable to requirements.
  2. The design implements proper sequence of events, inputs, outputs, interfaces, logic flow, allocation of timing and sizing budgets, and error definition, isolation, and recovery.
  3. Selected design can be derived from requirements.
  4. The design implements safety, security, and other critical requirements correctly as shown by suitably rigorous methods.
- 4. Code Verification:**
1. The code is traceable to design and requirements, testable, correct, and compliant with requirements and coding standards.
  2. 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.
  3. Selected code can be derived from design or requirements.
  4. The code implements safety, security, and other critical requirements correctly as shown by suitably rigorous methods.
- 5. Integration Verification:**
1. The software components and units of each software item have been completely and correctly integrated into the software item.
  2. The hardware items, software items, and manual operations of the system have been completely and correctly integrated into the system.
  3. The integration tasks have been performed in accordance with an integration plan.
- 6. Documentation Verification:**
1. The documentation is adequate, complete, and consistent.
  2. Documentation preparation is timely.
  3. Configuration management of documents follows specified procedures.

## 6.2.5 SOFTWARE VALIDATION PROCESS

### 6.2.5.1 PURPOSE

The purpose of the SOFTWARE VALIDATION PROCESS is to confirm that the requirements for a specific intended use of the software work product are fulfilled.

### 6.2.5.2 OUTCOMES

- a validation strategy is developed and implemented;
- criteria for validation of all required work products are identified;
- required validation activities are performed;
- problems are identified and recorded;
- evidence is provided that the software work products as developed are suitable for their intended use; and
- results of the validation activities are made available to the customer and other involved parties.

### 6.2.5.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. A determination shall be made if the project warrants a validation effort and the degree of organizational independence of that effort needed.
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.
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.
4. A validation plan shall be developed and documented. The plan shall include, but is not limited to, the following:
  1. Items subject to validation.
  2. Validation tasks to be performed.
  3. Resources, responsibilities, and schedule for validation.
  4. Procedures for forwarding validation reports to the acquirer and other parties.
5. The validation plan shall be implemented. Problems and non-conformances detected by the validation effort shall be entered into the SOFTWARE PROBLEM RESOLUTION PROCESS. All problems and non-conformances shall be resolved. Results of the validation activities shall be made available to the acquirer and other involved organizations.

#### 2. Validation:

1. Prepare selected test requirements, test cases, and test specifications for analyzing test results.
2. Ensure that these test requirements, test cases, and test specifications reflect the particular requirements for the specific intended use.
3. Conduct the tests in the previous two points, including:
  1. Testing with stress, boundary, and singular inputs;
  2. 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;
3. Testing that representative users can successfully achieve their intended tasks using the software product.
  4. Validate that the software product satisfies its intended use.
  5. Test the software product as appropriate in selected areas of the target environment.

## 6.2.6 SOFTWARE REVIEW PROCESS

### 6.2.6.1 PURPOSE

The purpose of the SOFTWARE REVIEW PROCESS is to maintain a common understanding with the stakeholders of the progress against the objectives of the agreement and what should be done to help ensure development of a product that satisfies the stakeholders.

Software reviews are at both project management and technical levels and are held throughout the life of the project.

### 6.2.6.2 OUTCOMES

- management and technical reviews are held based on the needs of the project;
- the status and products of an activity of a process are evaluated through review activities;
- review results are made known to all affected parties;
- action items resulting from reviews are tracked to closure; and
- risks and problems are identified and recorded.

### 6.2.6.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. Periodic reviews shall be held at predetermined milestones as specified in the project plan(s). Stakeholders should determine the need for any ad hoc reviews in which agreeing parties may participate.
2. All resources that are required to conduct the reviews shall be provided. These resources include personnel, location, facilities, hardware, software, and tools.
3. The parties that participate in a review 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.
4. Problems detected during the reviews shall be recorded and entered into the SOFTWARE PROBLEM RESOLUTION PROCESS as required.
5. The review results shall be documented and distributed. This communication includes adequacy of review (for example, approval, disapproval, or contingent approval) of the review results.
6. Participating parties shall agree on the outcome of the review and any action item responsibilities and closure criteria.

#### 2. Process Management Reviews:

1. Project status shall be evaluated relative to the applicable project plans, schedules, standards, and guidelines. The outcome of the review should be considered by appropriate management and should provide for the following:
  1. Making activities progress according to plan, based on an evaluation of the activity or software product status.
  2. Maintaining global control of the project through adequate allocation of resources.
  3. Changing project direction or determining the need for alternate planning.
  4. Evaluating and managing the risk issues that may jeopardize the success of the project.

#### 3. Technical Reviews:

1. Technical reviews shall be held to evaluate the software products or services under consideration and provide evidence that:
  1. They are complete.
  2. They comply with their standards and specifications.
  3. Changes to them are properly implemented and affect only those areas identified by the CONFIGURATION MANAGEMENT PROCESS.
  4. They are adhering to applicable schedules.
  5. They are ready for the next planned activity.
  6. The development, operation, or maintenance is being conducted according to the plans, schedules, standards, and guidelines of the project.

## 6.2.7 SOFTWARE AUDIT PROCESS

### 6.2.7.1 PURPOSE

The purpose of the SOFTWARE AUDIT PROCESS is to independently determine compliance of selected products and processes with the requirements, plans and agreement, as appropriate.

### 6.2.7.2 OUTCOMES

- an audit strategy is developed and implemented;
- compliance of selected software work products and/or services or processes with requirements, plans and agreement is determined according to the audit strategy;
- audits are conducted by an appropriate independent party; and
- problems detected during an audit are identified and communicated to those responsible for corrective action, and resolution.

### 6.2.7.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. Auditing personnel shall not have any direct responsibility for the software products and activities they audit.
2. All resources required to conduct the audits shall be agreed by the parties. These resources include support personnel, location, facilities, hardware, software, and tools.
3. 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.
4. Problems detected during the audits shall be recorded and entered into the SOFTWARE PROBLEM RESOLUTION PROCESS as required.
5. 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. The parties shall agree on the outcome of the audit and any action item responsibilities and closure criteria.

#### 2. Software Audit:

1. Software audits shall be conducted to ensure that:
  1. As coded, software products (such as a software item) reflect the design documentation.
  2. The acceptance review and testing requirements prescribed by the documentation are adequate for the acceptance of the software products.
  3. Test data comply with the specification.
  4. Software products were successfully tested and meet their specifications.
  5. Test reports are correct and discrepancies between actual and expected results have been resolved.
  6. User documentation complies with standards as specified.
  7. Activities have been conducted according to applicable requirements, plans, and contract.
  8. The costs and schedules adhere to the established plans.

## 6.2.8 SOFTWARE PROBLEM RESOLUTION PROCESS

### 6.2.8.1 PURPOSE

The purpose of the SOFTWARE PROBLEM RESOLUTION PROCESS is to ensure that all discovered problems are identified, analyzed, managed and controlled to resolution.

### 6.2.8.2 OUTCOMES

- a problem management strategy is developed;
- problems are recorded, identified and classified;
- problems are analyzed and assessed to identify acceptable solution(s);
- problem resolution is implemented;
- problems are tracked to closure; and
- the status of all problems reported is known.

### 6.2.8.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. A problem resolution process shall be established for handling all problems (including non-conformances) detected in the software products and activities. The process shall comply with the following requirements:
  1. The process shall be closed-loop, 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.
  2. The process should contain 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.
  3. Analysis shall be performed to detect trends in the problems reported.
  4. 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.

#### 2. Problem Resolution:

1. When problems (including non-conformances) 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 closed-loop process described above: from detection of the problem, through investigation, analysis and resolution of the problem and its cause, and onto trend detection across problems.

### 6.3 SOFTWARE REUSE PROCESSES

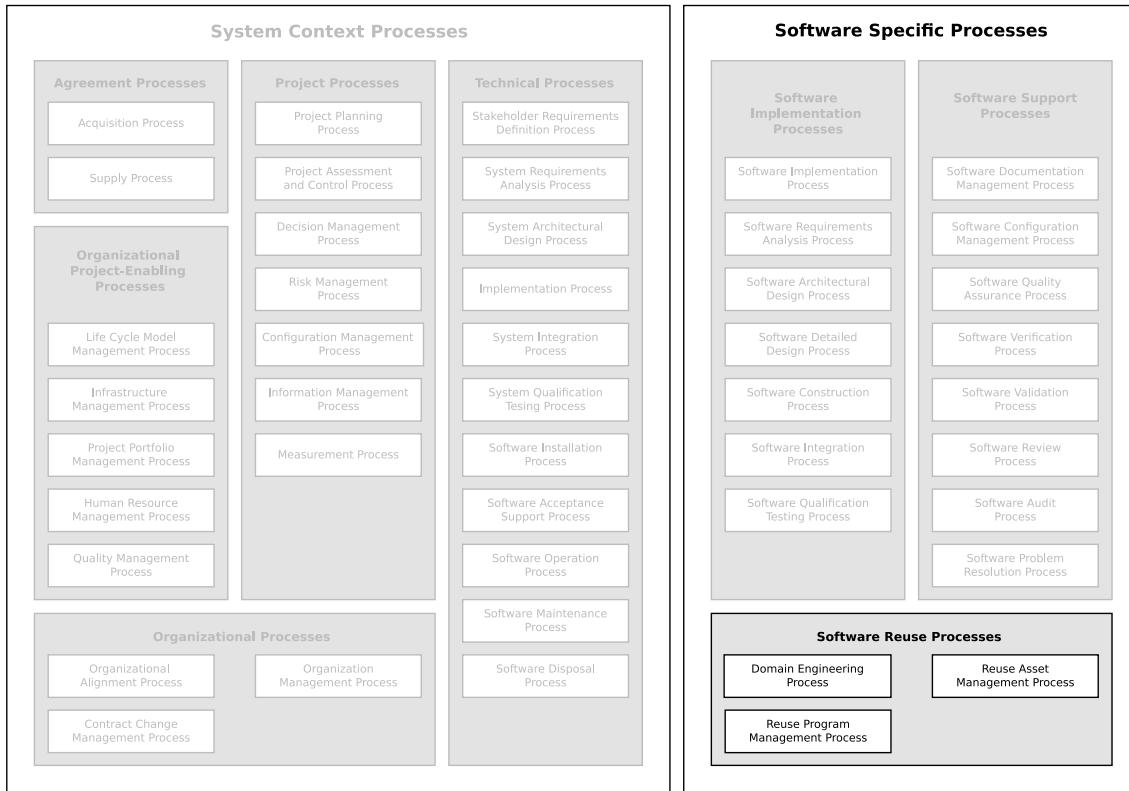


Figure 9: Software Reuse Processes

The SOFTWARE REUSE PROCESSES consist of three processes that support an organization's ability to reuse software items across project boundaries. These processes are unique because, by their nature, they operate outside the bounds of any particular project.

- 6.3.1 DOMAIN ENGINEERING PROCESS
- 6.3.2 REUSE ASSET MANAGEMENT PROCESS
- 6.3.3 REUSE PROGRAM MANAGEMENT PROCESS

### 6.3.1 DOMAIN ENGINEERING PROCESS

#### 6.3.1.1 PURPOSE

The purpose of the DOMAIN ENGINEERING PROCESS is to develop and maintain domain models, domain architectures and assets for the domain.

#### 6.3.1.2 OUTCOMES

- the representation forms for the domain models and the domain architectures are selected;
- the boundaries of the domain and its relationships to other domains are established;
- a domain model that captures the essential common and different features, capabilities, concepts, and functions in the domain are developed;
- a domain architecture describing the family of systems within the domain, including their commonalities and variabilities, is developed;
- assets belonging to the domain are specified;
- assets belonging to the domain are acquired or developed and maintained throughout their life cycles; and
- the domain models and architectures are maintained throughout their life cycles.

#### 6.3.1.3 ACTIVITIES AND TASKS

##### 1. Process Implementation:

1. The domain engineer shall create and execute a domain engineering plan.
2. The domain engineer shall select the form(s) of representation to be used for domain architectures and models.
3. The domain engineer shall establish procedures for receiving, resolving, and providing feedback to the asset manager whenever problems or change requests occur for assets developed by the domain engineer.

##### 2. Domain Analysis:

1. The domain engineer shall define the boundaries of the domain and the relationships between this domain and other domains.
2. The domain engineer shall identify the current and anticipated needs of stakeholders of software products within this domain.
3. The domain engineer shall build the domain models using the representation forms selected in the Process Implementation Activity for this process.
4. The domain engineer shall construct a vocabulary that provides the terminology to describe the important domain concepts and the relationships among similar or common assets of the domain.
5. The domain engineer shall classify and document the domain models.
6. The domain engineer shall evaluate the domain models and domain vocabulary in accordance with the provisions of the modeling technique selected and in accordance with the organizations asset acceptance and certification procedures.
7. The domain engineer shall conduct domain analysis review(s). Software developers, asset managers, domain experts, and users shall be included in the reviews.
8. The domain engineer shall submit domain models to the asset manager.

**3. Domain Design:**

1. The domain engineer shall create and document the domain architecture, consistent with the domain model and following the organizations standards.
2. The domain architecture shall be evaluated in accordance with the provisions of the architecture design technique selected and the organizations asset acceptance and certification procedures.
3. For each entity selected to be designed for reuse, the domain engineer shall develop and document an asset specification.
4. For each asset specified, the specification shall be evaluated in accordance with the organizations asset acceptance and certification procedures.
5. The domain engineer shall conduct domain design review(s). Software developers, domain experts, and asset managers shall be included in the reviews.
6. The domain engineer shall submit the domain architecture to the asset manager.

**4. Asset Provision:**

1. The domain engineer shall obtain the asset by acquisition or by development.
2. The domain engineer shall document and classify the asset.
3. The domain engineer shall evaluate the asset in accordance with the organizations asset acceptance and certification procedures.
4. The domain engineer shall conduct asset review(s). Software developers and asset managers shall be included in the reviews.
5. The domain engineer shall submit the asset to the asset manager.

**5. Asset Maintenance:**

1. When analyzing requests for asset modification and choosing implementation options, the domain engineer shall consider:
  1. Conformance with the domain models and the domain architecture;
  2. Impact on the systems and software products that use the asset;
  3. Impact on future users of the asset;
  4. Impact on the reusability of the asset.

## 6.3.2 REUSE ASSET MANAGEMENT PROCESS

### 6.3.2.1 PURPOSE

The purpose of the REUSE ASSET MANAGEMENT PROCESS is to manage the life of reusable assets from conception to retirement.

### 6.3.2.2 OUTCOMES

- an asset management strategy is documented;
- an asset classification scheme is established;
- criteria for asset acceptance, certification and retirement are defined;
- an asset storage and retrieval mechanism is operated;
- the use of assets is recorded;
- changes to the assets are controlled, and
- users of assets are notified of problems detected, modifications made, new versions created and deletion of assets from the storage and retrieval mechanism.

### 6.3.2.3 ACTIVITIES AND TASKS

#### 1. Process Implementation:

1. The asset manager shall create an asset management plan to define the resources and procedures for managing assets.
2. The asset manager shall execute the plan.
3. The asset management plan shall be reviewed in accordance with the Software Review Process. Domain engineers and reuse program administrators shall be included in the review.

#### 2. Asset Storage and Retrieval Definition:

1. The asset manager shall implement and maintain an asset storage and retrieval mechanism.
2. The asset manager should develop, document, and maintain a classification scheme to be used in classifying the assets.
3. The asset manager shall conduct review(s) of the asset storage and retrieval mechanism in accordance with the Software Review Process. Reuse program administrators and domain engineers shall be included in the review(s).

#### 3. Asset Management and Control:

1. For each asset submitted to the asset manager, the asset shall be evaluated based on the asset acceptance and certification criteria.
2. For each asset accepted, it shall be made available for reuse through the asset storage and retrieval mechanism.
3. The asset shall be classified in accordance with the reuse classification scheme, if any exists.
4. The asset manager shall perform configuration management for the asset using the Software Configuration Management Process.
5. The asset manager shall keep track of each reuse of the asset and report to the domain engineer information about actual reuses of the asset.
6. The asset manager shall forward asset modification requests and problem reports received from asset reusers to the domain engineer for review and correction/modification plans and actions.

7. The asset manager shall monitor and record these asset requests/reports and the subsequent actions taken.
8. The asset manager shall notify all asset reusers, and the domain engineer, of the problems detected in the asset, modifications made to the asset, new versions of the asset, and deletion of the asset from the asset storage and retrieval mechanism.
9. The asset manager shall retire assets from the asset storage and retrieval mechanism according to the asset retirement procedures and criteria.

### 6.3.3 REUSE PROGRAM MANAGEMENT PROCESS

#### 6.3.3.1 PURPOSE

The purpose of the REUSE PROGRAM MANAGEMENT PROCESS is to plan, establish, manage, control, and monitor an organization's reuse program and to systematically exploit reuse opportunities.

#### 6.3.3.2 OUTCOMES

- the organization's reuse strategy, including its purpose, scope, goals and objectives, is defined;
- the domains for potential reuse opportunities are identified;
- the organization's systematic reuse capability is assessed;
- the reuse potential of each domain is assessed;
- reuse proposals are evaluated to ensure the reuse product is suitable for the proposed application;
- the reuse strategy is implemented in the organization;
- feedback, communication, and notification mechanisms that operate between affected parties are established; and
- the reuse program is monitored and evaluated.

#### 6.3.3.3 ACTIVITIES AND TASKS

##### 1. Initiation:

1. The reuse program for an organization shall be initiated by establishing the organization's reuse strategy that includes its reuse goals, purposes, objectives, and scope.
2. A reuse sponsor should be named.
3. Reuse program participants shall be identified and their roles shall be assigned.
4. A reuse steering function shall be established to assume the authority and responsibility for the organization's reuse program.
5. A reuse program support function shall be established.

##### 2. Domain Identification:

1. The reuse program administrator, aided by the appropriate manager, domain engineers, users, and software developers, shall identify and document the domains in which to investigate reuse opportunities or in which the organization intends to practice reuse.
2. The reuse program administrator, aided by the appropriate managers, domain engineers, users, and software developers, shall evaluate the domains to assure that they accurately reflect the organization's reuse strategy.
3. The reuse program administrator shall conduct reviews in accordance with the Software Review Process. Software developers, domain engineers, and users shall be included in the reviews.
4. As more information about the organization's domains and plans for future software products becomes available or when the domains are analyzed, the domains may be refined and re-scoped by the reuse program administrator.

##### 3. Reuse Assessment:

1. The reuse program administrator shall assess the organization's systematic reuse capability.
2. The reuse program administrator shall assess each domain being considered for reuse to determine the potential for reuse success in the domain.

3. The reuse program administrator shall make recommendations for refining the organizations reuse strategy and reuse program implementation plan based on the results of the reuse assessments.
  4. The reuse program administrator, in conjunction with the appropriate acquirers, suppliers, developers, operators, maintainers, asset managers, and domain engineers, shall incrementally improve the skills, technology, reuse processes, organizational structure, and metrics that together comprise the reuse infrastructure.
- 4. Planning:**
1. A reuse program implementation plan shall be created, documented, and maintained to define the resources and procedures for implementing a reuse program.
  2. The plan shall be reviewed and evaluated for completeness, feasibility of implementation, and ability to realize the organization's reuse strategy. Those evaluating the plan should include members of the reuse steering function.
  3. Approval and support for the reuse program implementation plan shall be obtained from the reuse steering function, and the appropriate managers.
  4. The reuse program administrator shall conduct review(s) in accordance with the Software Review Process. Members of the reuse steering function and the appropriate managers shall be included in the reviews.
- 5. Execution and Control:**
1. The reuse program administrator shall monitor the progress of the reuse program against the organizations reuse strategy, and make any necessary adjustments to the plan to realize the strategy.
  2. Problems and non-conformances that occur during the execution of the reuse program implementation plan shall be recorded and resolved.
  3. The reuse program administrator shall periodically reaffirm management sponsorship, support, and commitment to the reuse program.
- 6. Review and Evaluation:**
1. The reuse program administrator shall periodically assess the reuse program for achievement of the organizations reuse strategy, and the continued suitability and effectiveness of the reuse program.
  2. The reuse program administrator shall provide assessment results and lessons learned to the reuse steering function, and to the appropriate managers.
  3. The reuse program administrator shall recommend and make changes to the reuse program, expand the reuse program, and improve the reuse program in accordance.

## A. LOWER-LEVEL PROCESSES

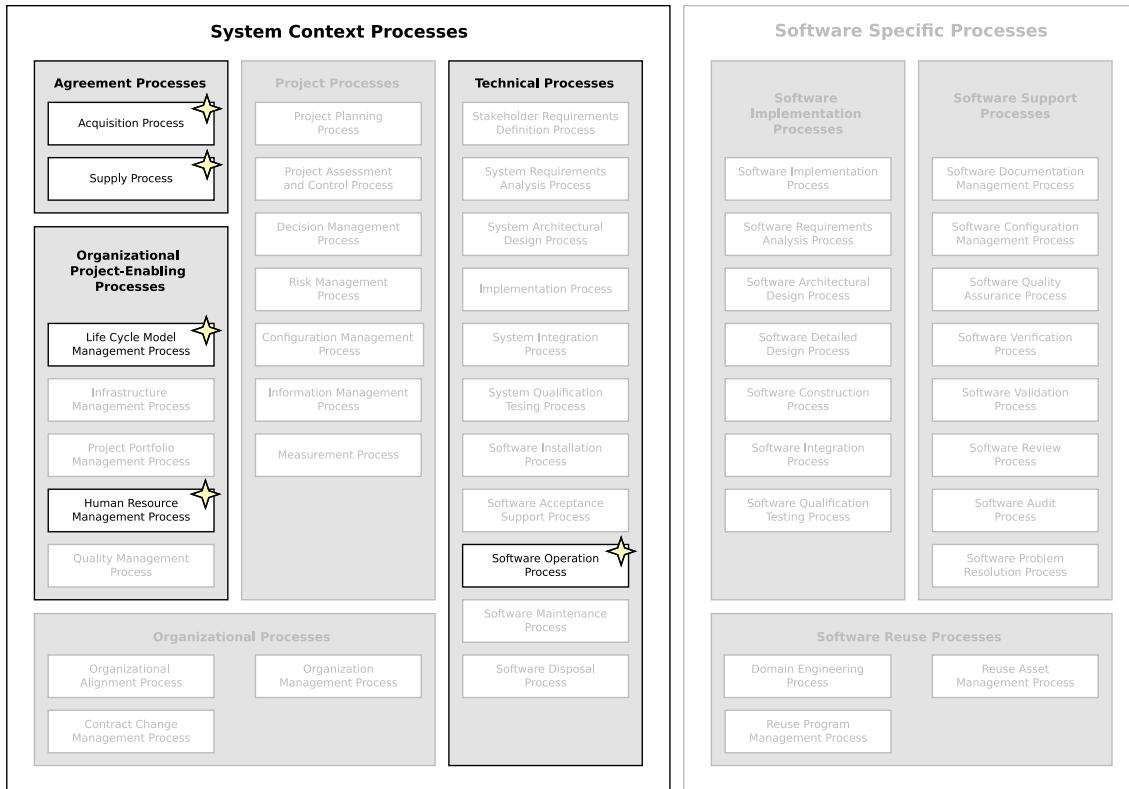


Figure 10: PROCESS REFERENCE MODEL (PRM) LOWER-LEVEL PROCESSES

Some of the processes defined within this standard may require additional lower-level processes in order to be realized in certain industries or by certain businesses whose activities are directed or more specific than the processes defined in the primary content of the document seemingly allow.

Only the purpose and outcomes for these processes has been defined, and the necessary activities for the processes, as they relate to those found in the parent-level process, need to be integrated into said process' activities and tasks in order to extend, and not disrupt, the parent-level process' outcomes, activities, and tasks.

## A.1 ACQUISITION PROCESS LOWER-LEVEL PROCESSES

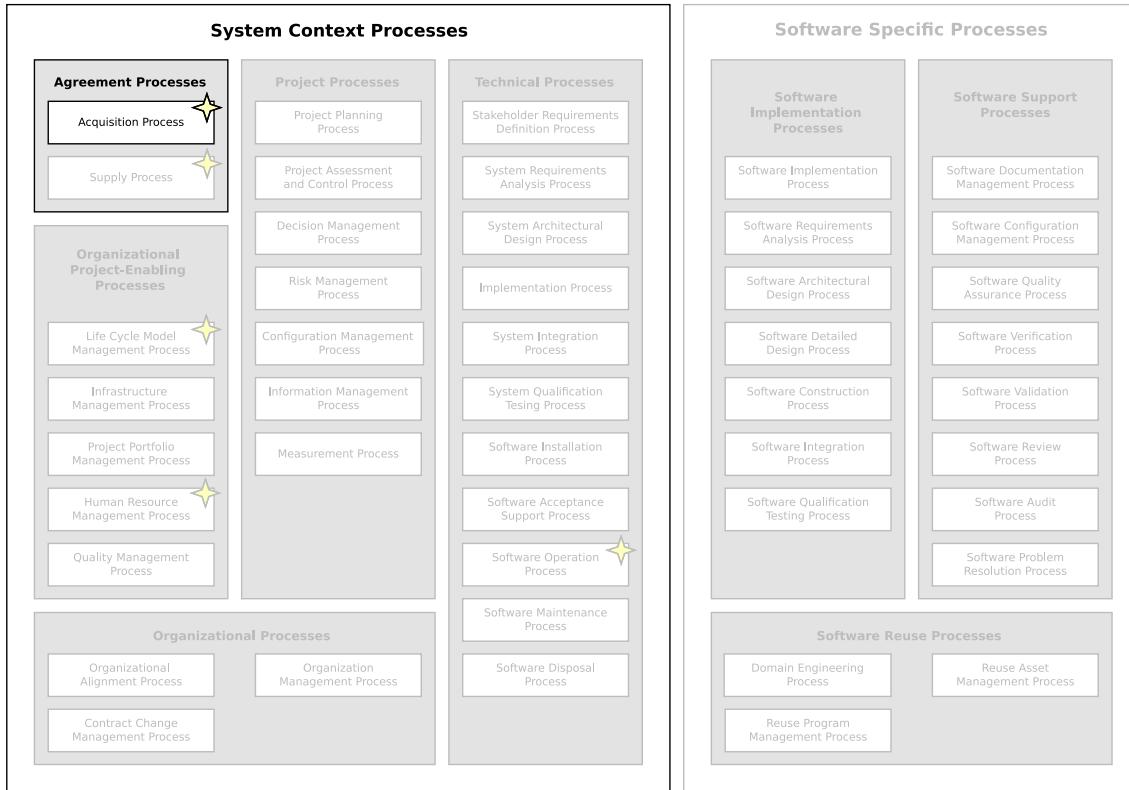


Figure 11: Acquisition Process Lower-Level Processes

### A.1.1 ACQUISITION PREPARATION PROCESS

#### A.1.1.1 PURPOSE

The purpose of the ACQUISITION PREPARATION PROCESS is to establish the needs and goals of the acquisition and to communicate these with the potential suppliers.

This process is a lower-level process of the ACQUISITION PROCESS, that replaces the Acquisition Preparation activity.

#### A.1.1.2 OUTCOMES

- the concept or the need for the acquisition, development, or enhancement is established;
- stakeholder requirements are defined;
- an acquisition strategy is developed; and
- supplier selection criteria are defined.

### A.1.2 SUPPLIER SELECTION PROCESS

#### A.1.2.1 PURPOSE

The purpose of the SUPPLIER SELECTION PROCESS is to choose the organization that is to be responsible for the delivery of the requirements of the project.

This process is a lower-level process of the ACQUISITION PROCESS, that replaces the Supplier Selection activity.

### A.1.2.2 OUTCOMES

- the supplier selection criteria are established and used to evaluate potential suppliers;
- the supplier is selected based upon the evaluation of the supplier's proposals, process capabilities, and other factors; and
- an agreement is established and negotiated between the acquirer and the supplier.

### A.1.3 AGREEMENT MONITORING PROCESS

#### A.1.3.1 PURPOSE

The purpose of the AGREEMENT MONITORING PROCESS is to track and assess performance of the supplier against agreed requirements.

This process is a lower level-process of the ACQUISITION PROCESS. It replaces the Agreement Monitoring activity.

#### A.1.3.2 OUTCOMES

- joint activities between the acquirer and the supplier are performed as needed;
- information on technical progress is exchanged regularly with the supplier;
- performance of the supplier is monitored against the agreed requirements; and
- agreement changes, if needed, are negotiated between the acquirer and the supplier and documented in the agreement.

### A.1.4 ACQUIRER ACCEPTANCE PROCESS

#### A.1.4.1 PURPOSE

The purpose of the ACQUIRER ACCEPTANCE PROCESS is to approve the supplier's deliverable when all acceptance criteria are satisfied.

This process is a lower-level process of the ACQUISITION PROCESS. It replaces the Acquirer Acceptance activity.

#### A.1.4.2 OUTCOMES

- the delivered software product and/or service are evaluated with regard to the agreement;
- the acquirer's acceptance is based on the agreed acceptance criteria; and
- the software product and/or service is accepted by the acquirer.

## A.2 SUPPLY PROCESS LOWER-LEVEL PROCESSES

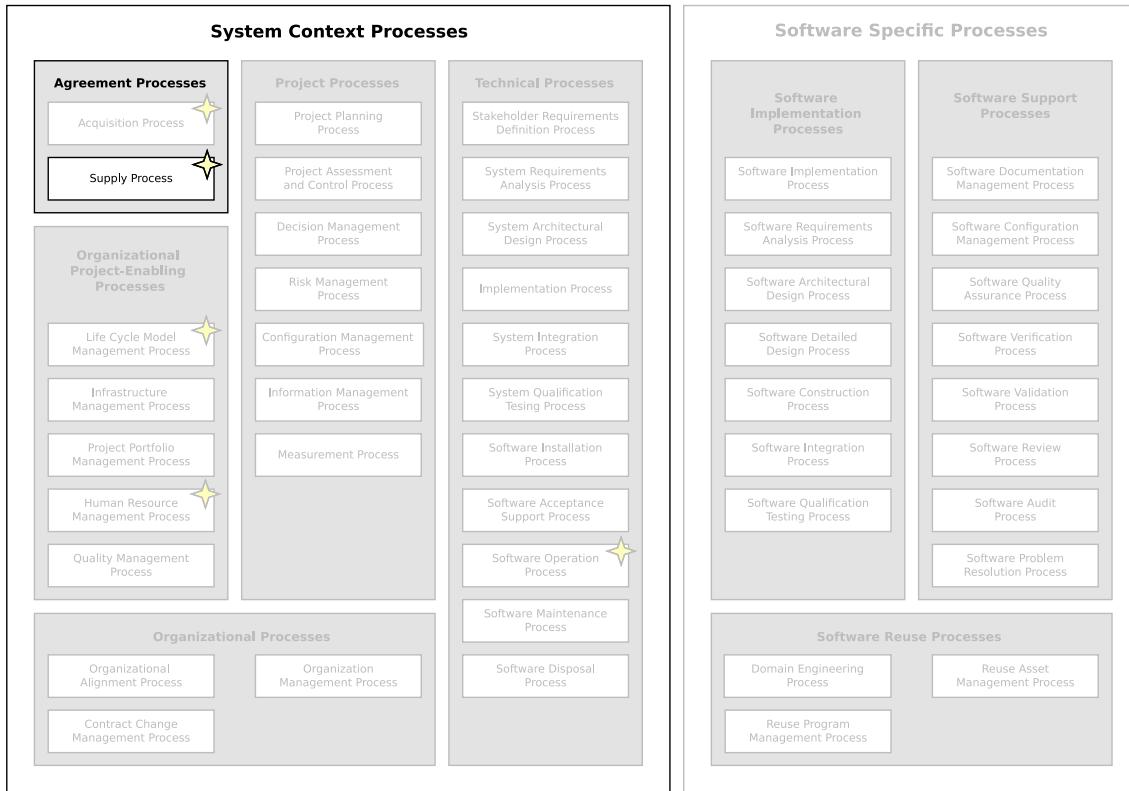


Figure 12: Supply Process Lower-Level Processes

### A.2.1 SUPPLIER TENDERING PROCESS

#### A.2.1.1 PURPOSE

The purpose of the SUPPLIER TENDERING PROCESS is to establish an interface to respond to acquirer inquiries and requests for proposal, prepare and submit proposals.

This process is a lower level-process of the SUPPLY PROCESS. It replaces the Supplier Tendering activity.

#### A.2.1.2 OUTCOMES

- a communication interface is established and maintained in order to respond to acquirer inquiries and requests for proposal;
- requests for proposal are evaluated according to defined criteria to determine whether or not to submit a proposal;
- the need to undertake preliminary surveys or feasibility studies is determined;
- suitable resources are identified to perform the proposed work; and
- a supplier proposal is prepared and submitted in response to the acquirer request.

## A.2.2 CONTRACT AGREEMENT PROCESS

### A.2.2.1 PURPOSE

The purpose of CONTRACT AGREEMENT PROCESS is to negotiate and approve a contract/agreement that clearly and unambiguously specifies the expectations, responsibilities, work products/deliverables and liabilities of both the supplier and the acquirer.

This process is a lower level-process of the SUPPLY PROCESS. It replaces the Contract Agreement activity.

### A.2.2.2 OUTCOMES

- a contract/agreement is negotiated, reviewed, approved and awarded to the supplier(s);
- mechanisms for monitoring the capability and performance of the supplier(s) and for mitigation of identified risks are reviewed and considered for inclusion in the contract conditions;
- proposers/tenderers are notified of the result of proposal/tender selection; and
- formal confirmation of agreement is obtained.

**Note:** The CONTRACT AGREEMENT PROCESS is used to obtain formal confirmation of assignments that were offered during the SUPPLIER TENDERING PROCESS.

## A.2.3 PRODUCT AND SERVICE DELIVERY AND SUPPORT PROCESS

### A.2.3.1 PURPOSE

The purpose of the PRODUCT AND SERVICE DELIVERY AND SUPPORT PROCESS is to provide the specified product or service to the acquirer with support appropriate to achieve confidence that the requirements have been met.

This process is a lower-level process of the SUPPLY PROCESS. It replaces the Product/Service Delivery and Support activity.

### A.2.3.2 OUTCOMES

- the contents of the product release are determined;
- the release is assembled from configured items;
- the release documentation is defined and produced;
- the release delivery mechanism and media are determined;
- release approval is effected against defined criteria;
- the product release is made available to the acquirer;
- confirmation of release is obtained;
- the product is completed and delivered to the acquirer;
- acquirer acceptance tests and reviews are supported;
- the product is put into operation in the customers environment; and
- problems detected during acceptance are identified and communicated to those responsible for resolution.

**Note:** Incremental delivery would be in completed increments.

### A.3 LIFE CYCLE MODEL MANAGEMENT PROCESS LOWER-LEVEL PROCESSES

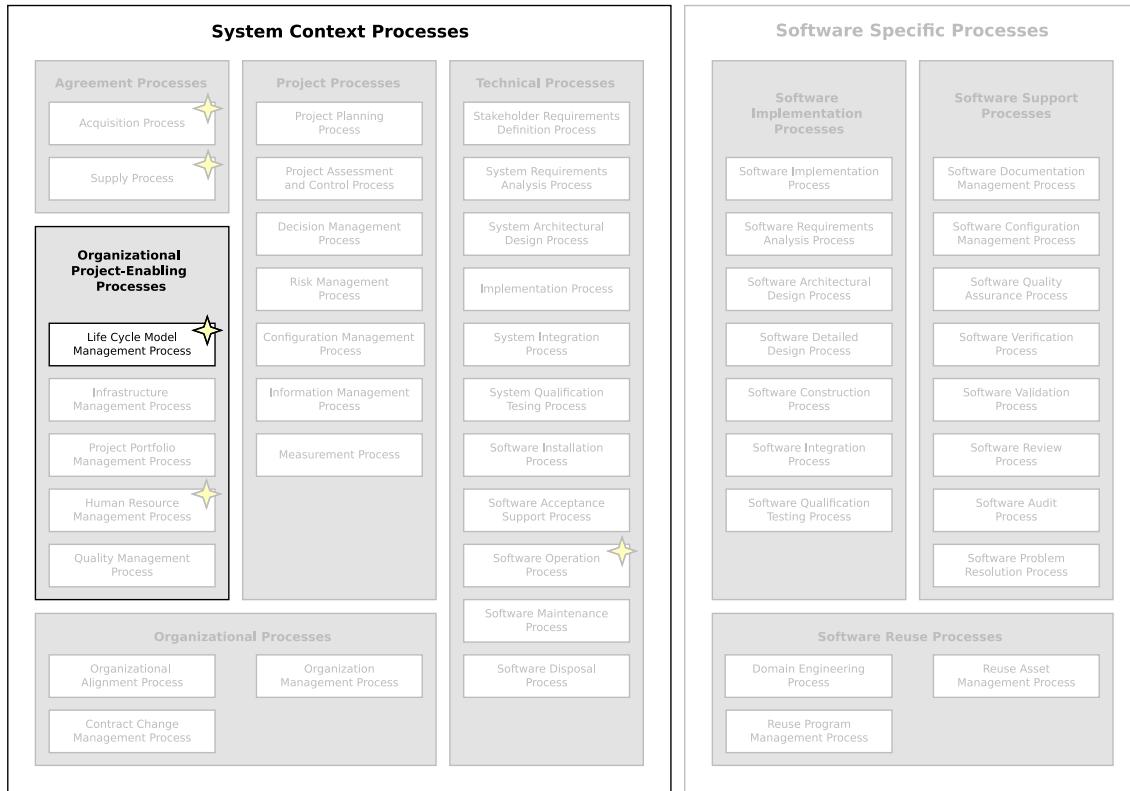


Figure 13: Life Cycle Model Management Process Lower-Level Processes

#### A.3.1 PROCESS ESTABLISHMENT PROCESS

##### A.3.1.1 PURPOSE

The purpose of the PROCESS ESTABLISHMENT PROCESS is to establish a suite of organizational processes for all life cycle processes as they apply to its business activities.

This process is a lower-level process of the LIFE CYCLE MODEL MANAGEMENT PROCESS. It replaces the Process Establishment activity.

##### A.3.1.2 OUTCOMES

- a defined and maintained standard set of processes are established, along with an indication of each process's applicability;
- the detailed tasks, activities and associated work products of the standard process are identified, together with expected performance characteristics;
- a strategy for tailoring the standard process for the product or service is developed in accordance with the needs of the project; and
- information and data related to the use of the standard process for specific projects exist and are maintained.

### A.3.2 PROCESS ASSESSMENT PROCESS

#### A.3.2.1 PURPOSE

The purpose of the PROCESS ASSESSMENT PROCESS is to determine the extent to which the organization's standard processes contribute to the achievement of its business goals and to help the organization focus on the need for continuous process improvement.

This process is a lower-level process of the LIFE CYCLE MODEL MANAGEMENT PROCESS, that replaces its Process Assessment activity.

#### A.3.2.2 OUTCOMES

- information and data related to the use of the standard process for specific projects exists and is maintained;
- the relative strengths and weaknesses of the organization's standard processes are understood; and
- accurate and accessible assessment records are kept and maintained.

### A.3.3 PROCESS IMPROVEMENT PROCESS

#### A.3.3.1 PURPOSE

The purpose of the PROCESS IMPROVEMENT PROCESS is to continually improve the organization's effectiveness and efficiency through the processes used and maintained and aligned with the business need.

This process is a lower-level process of the LIFE CYCLE MODEL MANAGEMENT PROCESS, that replaces its Process Improvement activity.

#### A.3.3.2 OUTCOMES

- commitment is established to provide resources to sustain improvement actions;
- issues arising from the organization's internal/external environment are identified as improvement opportunities and justified as reasons for change;
- analysis of the current status of the existing process is performed, focusing on those processes from which improvement stimuli arise;
- improvement goals are identified and prioritized, and consequent changes to the process are defined and implemented;
- the effects of process implementation are monitored and confirmed against the defined improvement goals;
- knowledge gained from the improvements is communicated within the organization; and
- the improvements made are evaluated and consideration given for using solutions elsewhere within the organization.

**Note 1:** Information sources providing input for change may include: process assessment results, audits, customer's satisfaction reports, organizational effectiveness/efficiency, cost of quality.

**Note 2:** The current status of processes may be determined by process assessment.

## A.4 HUMAN RESOURCE MANAGEMENT PROCESS LOWER-LEVEL PROCESSES

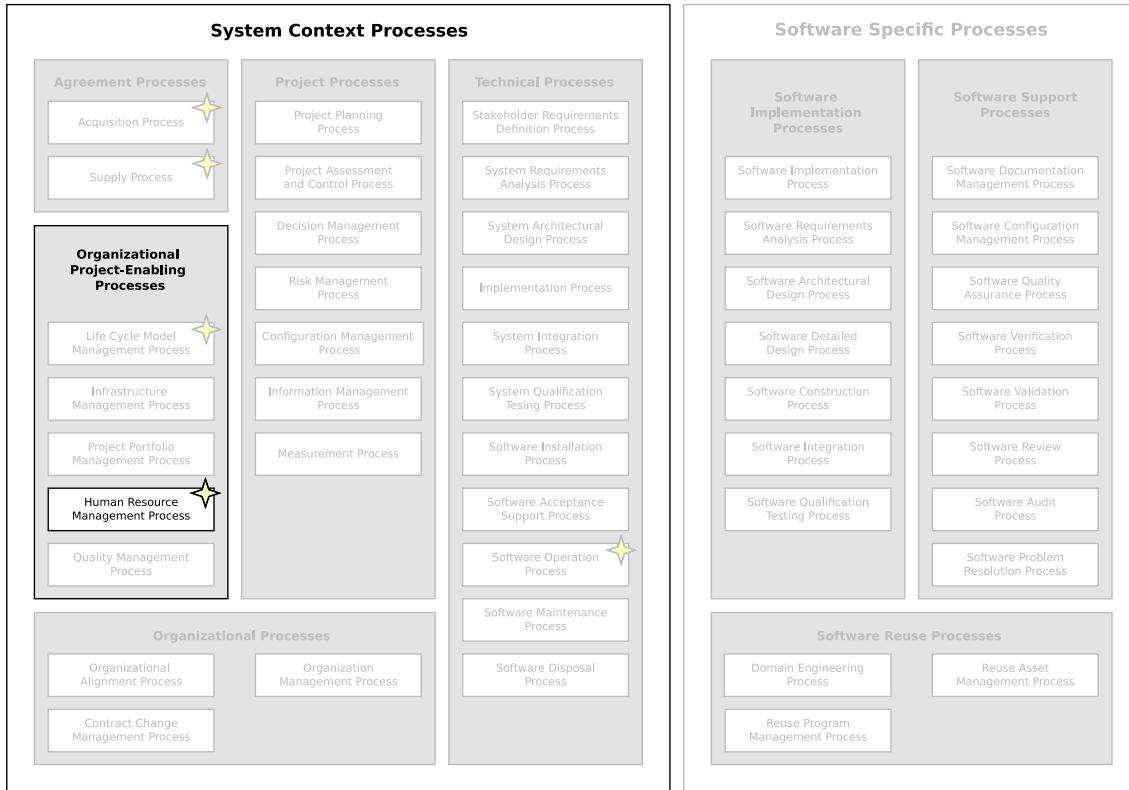


Figure 14: Human Resource Management Lower-Level Processes

### A.4.1 SKILL DEVELOPMENT PROCESS

#### A.4.1.1 PURPOSE

The purpose of the SKILL DEVELOPMENT PROCESS is to provide the organization and project with individuals who possess the needed skills and knowledge to perform their roles effectively.

This process is a lower-level process of the HUMAN RESOURCE MANAGEMENT PROCESS. It replaces the Skill Development activity.

#### A.4.1.2 OUTCOMES

- training is developed or acquired to address the organization and project training needs; and
- training is conducted to ensure that all individuals have the skills required to perform their assignments, using mechanisms such as training strategies and materials.

### A.4.2 SKILL ACQUISITION PROCESS

#### A.4.2.1 PURPOSE

The purpose of the SKILL ACQUISITION PROCESS is to provide the organization and projects with individuals who possess skills and knowledge to perform their roles effectively and to work together as a cohesive group.

This process is a lower-level process of the HUMAN RESOURCE MANAGEMENT PROCESS. It replaces the Skill Acquisition and Provision activity.

#### A.4.2.2 OUTCOMES

- individuals with the required skills and competencies are identified and recruited;
- effective interaction between individuals and groups are supported;
- the work force have the skills to share information and co-ordinate their activities efficiently; and
- objective criteria are defined against which group and individual performance is monitored to provide performance feedback and to enhance performance.

#### A.4.3 KNOWLEDGE MANAGEMENT PROCESS

##### A.4.3.1 PURPOSE

The purpose of the KNOWLEDGE MANAGEMENT PROCESS is to ensure that individual knowledge, information and skills are collected, shared, reused and improved throughout the organization.

This process is a lower-level process of the HUMAN RESOURCE MANAGEMENT PROCESS. It replaces the Knowledge Management activity.

##### A.4.3.2 OUTCOMES

- infrastructure is established and maintained for sharing common and domain information across the organization;
- knowledge is readily available and shared throughout the organization; and
- the organization selects an appropriate knowledge management strategy.

## A.5 SOFTWARE OPERATION PROCESS LOWER-LEVEL PROCESSES

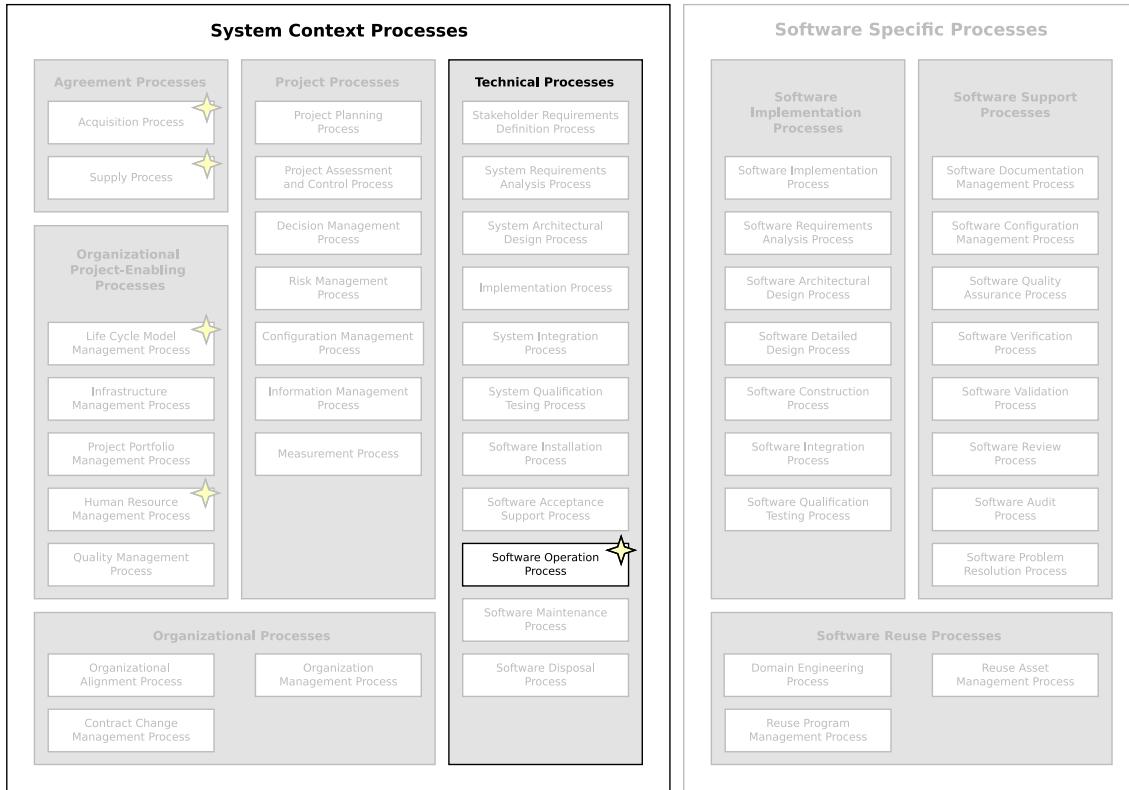


Figure 15: Software Operation Process Lower-Level Processes

### A.5.1 OPERATIONAL USE PROCESS

#### A.5.1.1 PURPOSE

The purpose of the OPERATIONAL USE PROCESS is to ensure the correct and efficient operation of the product for the duration of its intended usage and in its installed environment.

This process is a lower-level process of the SOFTWARE OPERATION PROCESS. It replaces the Operational Use activity.

#### A.5.1.2 OUTCOMES

- operational risks for the product introduction and operation are identified and monitored;
- the product is operated in its intended environment according to requirements; and
- criteria for the operational use are developed that demonstrates compliance with the agreed requirements.

### A.5.2 CUSTOMER SUPPORT PROCESS

#### A.5.2.1 PURPOSE

The purpose of the CUSTOMER SUPPORT PROCESS is to establish and maintain an acceptable level of service through assistance and consultation to the customer to support effective use of the product.

---

This process is a lower-level process of the SOFTWARE OPERATION PROCESS. It replaces the Customer Support activity.

#### A.5.2.2 OUTCOMES

- service needs for customer support are identified and monitored on an ongoing basis;
- customer satisfaction with both the support services being provided and the product itself is evaluated on an ongoing basis;
- operational support is provided by handling customer inquiries and requests and resolving operational problems; and
- customer support needs are met through delivery of appropriate services.

## B. PROCESS VIEWS

There are instances where those representing a particular engineering interest would like to see gathered in a single place the set of process activities that directly and succinctly address their concern. For such interests, a process view can be developed to organize processes, activities, and tasks selected from this and other standards to provide a focus to their particular concern in a manner that cuts across all or parts of the life cycle. This appendix provides a process viewpoint that may be used to define process views in these instances.

### B.1 THE PROCESS VIEW CONCEPT

There may be cases where a unified focus is needed for activities and tasks that are selected from disparate processes to provide visibility to a significant concept or thread that cuts across the processes employed across the life cycle. It is useful to advise users of the standards how to identify and define these activities for their use, even though they cannot locate a single process that addresses their specific concern.

For this purpose, the concept of a process view has been formulated. Like a process, the description of a process view includes a statement of purpose and outcomes. Unlike a process, the description of a process view does not include activities and tasks. Instead, the description includes guidance explaining how the outcomes can be achieved by employing the activities and tasks of the various processes in this standard. Process views can be constructed using the process viewpoint template found in the PROCESS VIEWPOINT sub-section below.

### B.2 PROCESS VIEWPOINT

A process view conforms to a process viewpoint. The process viewpoint provided here can be used to create process views. E.4 contains an example of applying this viewpoint.

The Process Viewpoint is defined by:

- its stakeholders: users of the standard;
- the concerns it frames: the processes needed to reflect a particular engineering interest;
- the contents of resulting process views should include:
  - process view name;
  - process view purpose;
  - process view outcomes; and
  - identification and description of the processes, activities and tasks which implement the process view, and references to the sources for these processes, activities and tasks in other standards.

### B.3 EXAMPLE: PROCESS VIEW FOR USABILITY

This section provides an example of applying the process viewpoint to yield a process view for Usability, intended to illustrate how a project might assemble processes, activities and tasks of this standard to provide focused attention to the achievement of a usable product.

This example treats the cluster of interests, generally called Usability, User centered design or Human-centered design that enables optimizing support and training, increased productivity and quality of work, improved human working conditions and reducing the chance of user rejection of the system.

**Name:** *Usability Process View*

### B.3.0.1 PURPOSE

The purpose of the Usability Process View is to ensure the consideration of the interests and needs of the stakeholders in order to enable optimizing support and training, increased productivity and quality of work, improved human working conditions and reducing the chance of user rejection of the system.

As a result of successful implementation of the *Usability Process View*:

- the system meets the needs of users and takes account of their human capabilities and skill limitations;
- human factors and ergonomics knowledge and techniques are incorporated in systems design;
- human-centered design activities are identified and performed;
- system design will address possible adverse effects of use on human health, safety and performance; and
- systems will have enhanced user effectiveness, efficiency and satisfaction.

**Note:** Although involvement of users is a principle of human centered design, the outcomes permit the possibility that the desired characteristics cannot be directly measured but instead might be argued and inferred based on other product or process characteristics that can be measured.

### B.3.0.2 PROCESSES, ACTIVITIES, TASKS

This process view can be implemented using the following processes, activities, and tasks from this standard.

1. The PROJECT PORTFOLIO MANAGEMENT PROCESS, in particular the Process Initiation activity, provides for the establishment and maintenance of a focus on user issues in the parts of the organization that deal with markets, concept, development and support; championing of a human-centered approach.
2. The INFRASTRUCTURE MANAGEMENT PROCESS provides a specification of how human-centered design activities fit into the whole systems life cycle process and the organization.
3. The PROJECT PLANNING PROCESS provides for: selection of human centered methods and techniques, planning the involvement of users and other stakeholders, planning of human-centered design activities.
4. The PROJECT ASSESSMENT AND CONTROL PROCESS provides for monitoring the extent of achievement of the requirements and communicating the results to stakeholders and managers, ensuring a human centered approach in the design team.
5. The STAKEHOLDER REQUIREMENTS DEFINITION PROCESS provides for the identification and documentation of the context of use and the interaction between users and the system, taking into account human capabilities and skills limitations and the specification of health, safety, security, environment, training, support and other stakeholder requirements and functions that address possible adverse effects of use of the system on human health and safety.
6. The SYSTEM REQUIREMENTS ANALYSIS PROCESS provides for the specification and evaluation of the context of use and the usability and human centered design requirements.
7. The SYSTEM ARCHITECTURAL DESIGN PROCESS provides for the incorporation of design criteria to address the targets for usability and the ergonomic requirements.
8. The SYSTEM INTEGRATION PROCESS provides for planning the integration, including the considerations for user training and the assurance that the achievement of targets for usability and accordance with ergonomic requirements are verified and recorded.
9. The INFORMATION MANAGEMENT PROCESS, in its entirety, provides for the specification, development and maintenance of artifacts for documenting and communicating the extent of achievement.

10. The MEASUREMENT PROCESS, in its entirety, provides for defining an approach that relates measures to desired characteristics.
11. SOFTWARE REQUIREMENTS ANALYSIS PROCESS provides for the specification of the usability and software ergonomics requirements.
12. The SOFTWARE OPERATION PROCESS provides for use of the system. Assuring that the usability requirements are appropriately achieved involves monitoring the operation of the system.
13. The SOFTWARE MAINTENANCE PROCESS sustains the capabilities of the system, including its usability properties and can be used in its entirety.

## C. RELATIONSHIPS BETWEEN STANDARDS

The purpose of this informative appendix is to describe relationships to other IEEE standards. The itemized breakdown below lists the processes of this standard, this document specifically. For many of those processes, the list suggests IEEE standards that may be helpful in implementing or executing the process. In each case, a note describes the nature of the relationship.

- AGREEMENT PROCESSES
  - ACQUISITION PROCESS: IEEE Std. 1062, recommendation of useful practices that can be selected and applied during software acquisition.
- ORGANIZATIONAL PROJECT-ENABLING PROCESSES
  - LIFE CYCLE MODEL MANAGEMENT PROCESS: IEEE Std. 1074, standard describes an approach for the definition of software life cycle processes.
  - INFRASTRUCTURE MANAGEMENT PROCESS : IEEE Std. 1175 and 1462, describe the integration of CASE tools into productive software engineering environments, and guidelines for the evaluation and selection of CASE tools, respectively.
  - QUALITY MANAGEMENT PROCESS: ISO 90003, standard provides guidance for organizations in the application of ISO 9001:2000 to software.
- PROJECT PROCESSES
  - PROJECT PLANNING PROCESS: IEEE Std. 1058 and 1228, describes the format and content of a software project management plan, and the content of a plan for the software aspects of development, procurement, maintenance, and retirement of a safety-critical system, respectively.
  - RISK MANAGEMENT PROCESS: IEEE Std. 1540, provides a process for the management of software risk.
  - MEASUREMENT PROCESS: IEEE Std. 982.1, 1045, 1061, and 14143.1, provides a set of measures for forecasting and evaluating the reliability of a software product, a consistent terminology for software productivity measures, a methodology for establishing quality requirements, and the fundamental concepts of a class of measures collectively known as functional size, respectively.
- TECHNICAL PROCESSES
  - STAKEHOLDER REQUIREMENTS DEFINITION PROCESS: IEEE Std. 1362, provides guidance on the format and content of a Concept of Operations document, describing characteristics of a proposed system from the user's viewpoint.
  - SYSTEM REQUIREMENTS ANALYSIS PROCESS: IEEE Std. 1233, provides guidance on the development of a system requirements specification and the characteristics and qualities of requirements.
  - SYSTEM ARCHITECTURAL DESIGN PROCESS, IEEE Std. 1471, recommends a conceptual framework and content for the architectural description of software-intensive systems.
  - SOFTWARE MAINTENANCE PROCESS: ISO/IEC 14764, provides guidance on implementing the software maintenance process.
- SOFTWARE IMPLEMENTATION PROCESSES
  - SOFTWARE REQUIREMENTS ANALYSIS PROCESS: IEEE Std. 830, recommends the content and characteristic of a software requirements specification.
  - SOFTWARE ARCHITECTURAL DESIGN PROCESS: IEEE Std. 1471, recommends a conceptual framework and content for the architectural description of software-intensive systems.
  - SOFTWARE DETAILED DESIGN PROCESS: IEEE Std. 1016, recommends content and organization of a software design description.
  - SOFTWARE CONSTRUCTION PROCESS: IEEE Std. 1008, describes an approach to software unit

testing.

- SOFTWARE INTEGRATION PROCESS: IEEE 829, describes the form and content of a basic set of documentation for planning, executing, and reporting software testing.
- SOFTWARE QUALIFICATION TESTING PROCESS: IEEE 829, describes the form and content of a basic set of documentation for planning, executing, and reporting software testing.
- SOFTWARE SUPPORT PROCESSES
  - SOFTWARE DOCUMENTATION MANAGEMENT PROCESS: IEEE Std. 1063, provides requirements for the structure, content, and format of user documentation.
  - SOFTWARE CONFIGURATION MANAGEMENT PROCESS: IEEE Std. 828, specifies the content of a software configuration management plan along with requirements for specific planning activities.
  - SOFTWARE QUALITY ASSURANCE PROCESS: IEEE Std. 730, 1061, and 1465, specifies the format and content of a software quality assurance plan, a methodology for establishing quality requirements and for identifying, implementing, and validating the corresponding measures, and describes quality requirements specifically suitable for software packages, respectively.
  - SOFTWARE VERIFICATION PROCESS: IEEE Std. 1012, describes software verification and validation activities.
  - SOFTWARE VALIDATION PROCESS: IEEE Std. 1012, describes software verification and validation activities.
  - SOFTWARE REVIEW PROCESS: IEEE Std. 1028, describes five types of software reviews, and procedures for their execution.
  - SOFTWARE AUDIT PROCESS: IEEE Std. 1028, describes five types of software reviews, and procedures for their execution.
  - SOFTWARE PROBLEM RESOLUTION PROCESS: IEEE Std. 1044, provides a uniform approach to the classification of anomalies found in software and its documentation.
- SOFTWARE REUSE PROCESSES
  - For all processes within this section, IEEE Standard 1420.1 and 1517

## D. TAILORING PROCESS

### D.1 PURPOSE OF THE TAILORING PROCESS

The purpose of the Tailoring Process is to adapt the processes of this standard to satisfy particular circumstances or factors that:

1. Surround an organization that is employing this standard.
2. Influence a project that is required to meet an agreement.
3. Reflect the needs of an organization in order to supply products or services.

### D.2 OUTCOMES OF THE TAILORING PROCESS

As a result of the successful implementation of the Tailoring Process:

1. Identify and document the circumstances that influence tailoring. These influences include, but are not limited to:
  1. Stability of, and variety in, operational environments.
  2. Risks, commercial or performance, to the concern of interested parties.
  3. Novelty, size and complexity.
  4. Starting date and duration of utilization.
  5. Integrity issues such as safety, security, privacy, usability, availability.
  6. Emerging technology opportunities.
  7. Profile of budget and organizational resources available.
  8. Availability of the services of enabling systems.
  9. Roles and responsibilities in the overall life cycle of the system.
  10. The need to conform to other standards.
2. In the case of properties critical to the system, take due account of the life cycle structures recommended or mandated by standards relevant to the dimension of the criticality.
3. Obtain input from all parties affected by the tailoring decisions. This includes, but may not be limited:
  1. The system stakeholders.
  2. The interested parties to an agreement made by the organization.
  3. The contributing organizational functions.
4. Make tailoring decisions in accordance with the DECISION MANAGEMENT PROCESS to achieve the purposes and outcomes of the selected life cycle model.
5. Select the life cycle processes that require tailoring and delete selected outcomes, activities, or tasks.

### D.3 ACTIVITIES OF THE TAILORING PROCESS

As a result of the successful implementation of the Tailoring Process:

1. Modified life cycle processes are defined to achieve the purposes and outcomes of a life cycle model.

#### Notes:

- Organizations establish standard life cycle models as a part of the LIFE CYCLE MODEL MANAGEMENT PROCESS. It may be appropriate for an organization to tailor processes of this standard in order to achieve the purposes and outcomes of the stages of a life cycle model to be established.
- Projects select an organizationally-established life cycle model for the project as a part of the PROJECT PLANNING PROCESS. It may be appropriate to tailor organizationally-adopted processes to achieve the

purposes and outcomes of the stages of the selected life cycle model.

- In cases where projects are directly applying this standard, it may be appropriate to tailor processes of this standard in order to achieve the purposes and outcomes of the stages of a suitable life cycle model.
- Irrespective of tailoring, organizations and projects are always permitted to implement processes that achieve additional outcomes or implement additional activities and tasks beyond those required for conformance to this standard.
- An organization or project may encounter a situation where there is the desire to modify a provision of this standard. Modification should be avoided because it may have unanticipated consequences on other processes, outcomes, activities or tasks. If necessary, modification is performed by deleting the provision (making the appropriate claim of tailored conformance) and, with careful consideration of consequences, implementing a process that achieves additional outcomes or performs additional activities and tasks beyond those of the tailored standard.

