ISO/IEC JTC 1 N 9262

2008-09-08

ISO/IEC JTC 1
Information Technology

Document Type:	**Text for DTR Ballot**
Document Title:	**DTR 24748.2 Systems and software engineering — Life cycle management — Guide for life cycle management**
Document Source:	**JTC 1 Secretariat**
Reference:	
Document Status:	**This document is circulated to JTC 1 National Bodies for a 3 month DTR ballot. National Bodies are asked to vote and submit their comments via the on-line balloting system by the due date indicated.**
Action ID:	**VOTE**
Due Date:	**2008-12-09**
No. of Pages:	**89**

Secretariat, ISO/IEC JTC 1, American National Standards Institute, 25 West 43rd Street, New York, NY 10036; Telephone: 1 212 642 4932; Facsimile: 1 212 840 2298; Email: lrajchel@ansi.org

ISO/IEC JTC1/SC7
Software and Systems Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 /N4143

2008-09-07

Document Type	DTR
Title	DTR 24748.2 Systems and software engineering — Life cycle management — Guide for life cycle management
Source	SC7 Secretariat
Project	
Status	DTR
Reference	SC7 Resolution 1047, N3892, N4013, N4142
Action ID	ACT or INF
Distribution	AG, JTC 1 Secretariat
No. of Pages	87
Note	Sent to the JTC 1 Secretariat for DTR processing

Address reply to: ISO/IEC JTC1/SC7 Secretariat
École de technologie supérieure – Department of Software and IT Engineering
1100 Notre Dame Ouest, Montréal, Québec Canada H3C 1K3
secretariat@jtc1-sc7.org

www.jtc1-sc7.org

ISO/IEC JTC 1/SC 7

Date: 2008-08-11

ISO/IEC DTR 24748-1

ISO/IEC JTC 1/SC 7/WG 7 **N1140**

Secretariat: SCC

Systems and software engineering — — Guide for life cycle management

Ingénierie systémes et logiciel — — Guide pour gestión du cycle de vie

It is the intention of this project to create a Technical Report of Type 3 that may be made freely available in accordance with the provisions of JTC 1 N 7269 and Sendai Resolution 32. In particular, the document has the following characteristics:

- it describes a reference model (Criterion 5) ;
- it explains the relationships between ISO/IEC 12207 and ISO/IEC 15288 (Criterion 5);
- this Part 1 provides a framework to guide development of the two application guides for ISO/IEC 15288:2008 and ISO/IEC 12207:2008, which will be for-sale Parts 2 and 3, respectively (Criterion 6);
- it describes the basic visions and concepts in the domains covered by life cycle standards, among which are ISO/IEC 12207, ISO/IEC 15288, (Criterion 8)
- it gives references to several other standards;
- it serves to explain the inter-relationships among these standards;
- it serves to promote the work of JTC1; and
- it is expected to increase the sales of the referenced standards.

Document type: Technical Report
Document subtype:
Document stage: (50) ApprovalEnquiry
Document language: E

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, the joint technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard (“state of the art”, for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 3, which is a Technical Report of type 3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Systems and software engineering*.

Introduction

International Standards ISO/IEC 15288 (Systems and software engineering—System life cycle processes) and ISO/IEC 12207 (Systems and software engineering—Software life cycle processes) each have published guides (ISO/IEC TR 15271 and ISO/IEC TR 19760, respectively) for the use of each standard individually. The purpose of this Technical Report is to facilitate the joint usage of the process content of the latest revisions of those documents by providing unified and consolidated guidance on life cycle management of systems and software. This is to help ensure consistency in system concepts and life cycle concepts, models, stages, processes, process application, key points of view, adaptation and use in various domains as the two standards are used in combination. That will in turn help a project design a life cycle model for managing the progress of their project.

The Technical Report will also aid in identifying and planning use of life cycle processes described in ISO/IEC 12207 and ISO/IEC 15288 that will enable the project to be completed successfully, meeting its objectives/requirements for each stage and for the overall project. It is planned that the other two guides (ISO/IEC TR 15271 and ISOIEC TR 19760) will be modified to support use of the two revised International Standards individually.

Besides the above, there is also increasing recognition of the importance of ensuring that all life cycle stages, and all aspects within each stage, are supported with thorough guidance to enable alignment with any process documents that may subsequently be created that focus on areas besides systems and software, including hardware, humans, processes (e.g. review process), procedures (e.g. operator instructions), facilities and naturally occurring entities (e.g. water, organisms, minerals).

By addressing these needs specifically in this Technical Report, the users of the process-focused ISO/IEC 12207 and ISO/IEC 15288 will not only benefit from having one document complementarily addressing the aspect of product or service life cycle: they will also benefit from a framework that links life cycle management aspects to more than just the systems or software aspects of products or services.

Systems and software engineering — — Guide for life cycle management

1 Scope

This Technical Report is a guide for the life cycle management of systems and software based on ISO/IEC 15288 and ISO/IEC 12207. This guide:

- addresses systems concepts and life cycle concepts, models, stages, processes, process application, key points of view, adaptation and use in various domains
- establishes a common framework for describing life cycles, including their individual stages, for the management of projects to provide or acquire either products or services
- defines the concept and terminology of a life cycle
- supports the use of the life cycle processes within an organization or a project. Organizations and projects can use these life cycle concepts when acquiring and supplying either products or services
- provides guidance on adapting a life cycle model and the content associated with a life cycle or a part of a life cycle
- describes the relationship between life cycles and their use in ISO/IEC 15288 (systems aspects) and ISO/IEC 12207 (software aspects)
- shows the relationships of life cycle concepts to the hardware, human, services, process, procedure, facility and naturally occurring entity aspects of projects
- describes how its concepts relate to detailed process standards, for example, in the areas of measurement, project management and risk management
- complements domain-specific application guidance in Technical Reports ISO/IEC TR 15271 (Information technology—Guide for ISO/IEC 12207) and ISO/IEC TR 19760 (Systems engineering—Guide for the application of ISO/IEC 15288).

2 Terms and definitions

This technical report uses the same terms and definitions that are used in ISO/IEC 12207 and ISO/IEC 15288, whether the terms were defined in those two International Standards or were defined in other International Standards. This technical report does not define any additional terms beyond that. The full set of terms so defined is found in Annex A.

3 Life cycle-related concepts

3.1 System concepts

3.1.1 Introduction

This section is included to highlight and to help explain essential concepts on which this Technical Report is based. These concepts are directly applicable to software, as addressed in ISO/IEC 12207, systems, as addressed in ISO/IEC 15288, and to hardware, facilities, services, humans, processes and procedures and naturally occurring entities.

3.1.2 Systems

The systems considered in this Technical Report are man-made and utilized to provide services in defined environments for the benefit of users and other stakeholders. These systems may be configured with one or more of the following: hardware, software, services, humans, processes (e.g. review process), procedures (e.g. operator instructions), facilities and naturally occurring entities (e.g. water, organisms, minerals).

The perception and definition of a particular system, its architecture and its system elements depend on an observer’s interests and responsibilities. One person's system-of-interest can be viewed as a system element in another person's system-of-interest. Conversely, it can be viewed as being part of the environment of operation for yet another person's system-of-interest.

Figure 1 exemplifies the multitude of perceivable systems-of-interest in an aircraft and its environment of operation. First, the figure in its entirety comprises a transportation system with air, ground and water elements, which can be a system-of-interest. Next, any one element of the transportation system can be viewed as a system-of-interest, such as the Air Transport System. The example can be continued through the levels so that, for example, the display can be an element of the navigation system, which is in turn an element of the aircraft system within the air transport system. However, from an equally valid perspective, such as that of display

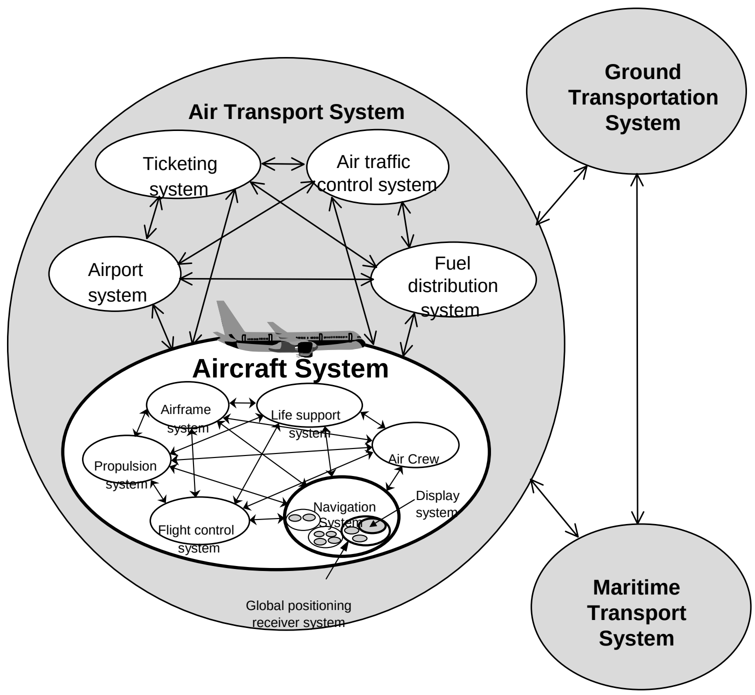

Figure 1—Typical system view of an aircraft in its environment of use

manufacturers, the display will be their system-of-interest and they will then determine the elements within their display system. On a deeper level, the Figure also illustrates:

a) the importance of defined boundaries that encapsulate meaningful needs and practical solutions;

b) the perception of system structure (in this case, the physical structure, hierarchical in this instance);

c) that an entity at any level in a system structure can be viewed as a system;

d) that a system is comprised of a fully integrated, defined set of subordinate systems;

e) that characteristic properties at a system's boundary arise from the interactions between subordinate systems;

f) that humans can be viewed as users external to a system (e.g. air crew and navigation system) and as system elements within a system (e.g. air crew and aircraft);

g) that a system can be viewed as an isolated entity (that is, a product), or as an ordered collection of functions capable of interacting with its surrounding environment, (i.e. a set of services);

h) that a system can be viewed as part of a larger system of systems and that a system of systems can be viewed as a system.

Whatever the boundaries chosen to define the system, the concepts and models in this Technical Report are generic and permit a practitioner to correlate or adapt individual instances of life cycles to its system concepts and principles.

In this Technical Report, humans are considered both as users and as elements of a system. In the first case, the human user is a beneficiary of the operation of the system. In the second case, the human is an operator carrying out specified system functions, such as those involved in providing a service. An individual can be, simultaneously or sequentially, a user and an element of a system, for example, the pilot of a private aircraft in the civil aviation system.

Humans contribute to the performance and characteristics of many systems for numerous reasons, e.g. their special skills, the need for flexibility and for legal reasons. Whether they are users or operators, humans are highly complex, with behaviour that is frequently difficult to predict, and they need protection from harm. This requires that the system life cycle processes address human element factors in the areas of: human factors engineering, system safety, health hazard assessment, manpower, personnel and training. These issues are addressed by particular activities and iterations in the life cycle, and are described in more detail in ISO 13407:1999, Human-centred design processes for interactive systems, and ISO/TR 18529:2000, Ergonomics—Ergonomics of human-system interaction—Human-centered lifecycle process descriptions.

3.1.3 System structure

The system life cycle processes in this Technical Report are described in relation to a system that is composed of a set of interacting physical, logical and/or other system elements, depicted in Figure 2, each of which can be implemented to fulfil its respective specified requirements. Responsibility for the implementation of any system element may therefore be delegated to another party through an agreement.

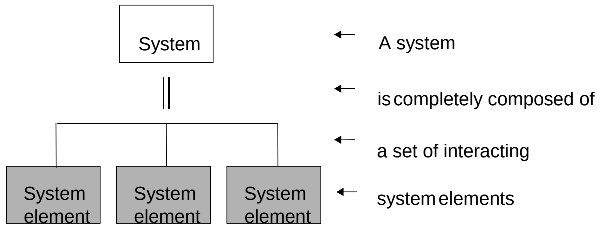

Figure 2—System and system element relationship

The relationship between the system and its complete set of system elements can typically be resolved in a single step only for the simplest of systems-of-interest. For more complex systems-of-interest, a prospective system element may itself need to be considered as a system (that in turn is comprised of system elements) before a complete set of system elements can be defined with confidence, as indicated by Figure 3. In this manner, the system life cycle processes are applied recursively to a system-of-interest to resolve its structure to the point where understandable and manageable system elements can be implemented or reused, or acquired from another party. While Figures 2 and 3 imply a hierarchical relationship, in reality there are an increasing number of systems that, from one or more aspects are not hierarchical, such as networks and other distributed systems. So, recursion is not necessarily linearly downward in all cases.

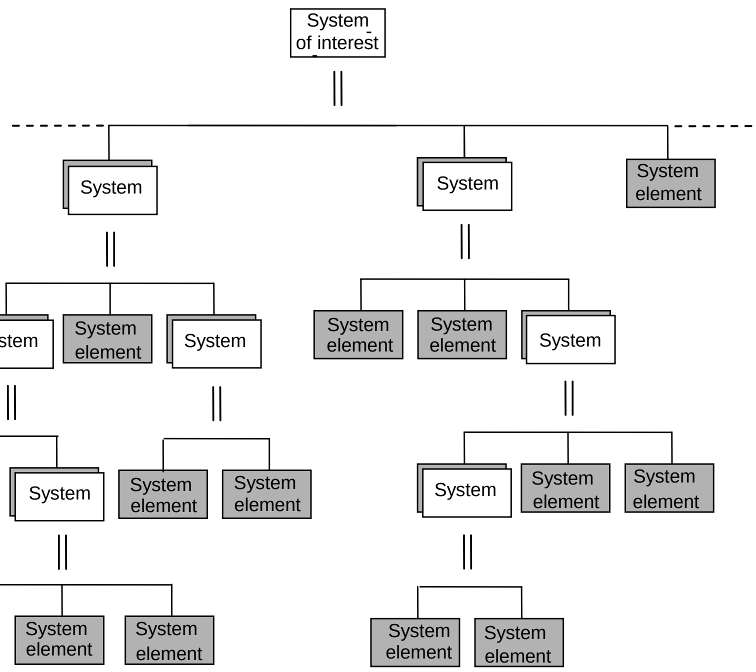

Figure 3—System-of-interest structure

3.1.4 Structure in systems and projects

Each system in the structure illustrated in Figure 2 (or Figure 3) could be the responsibility of a separate project. This can be true whether the system is hierarchical, as used for the illustration, or other system structure, such as a network, or a mix of structures. The point is that there can be (and typically is) a strong correlation between levels of detail in the system structure and levels of responsibility in a set of projects. Each project characteristically has responsibility for acquiring and using system elements subordinate to it and creating and supplying to the level superior to it, as would be the case for hierarchical, non-hierarchical, or mixed, system structures.

Any particular project normally views its system as the system-of-interest, and, whilst it can influence higher system levels, it does not have responsibility for them. However, even though it may not have responsibility for each system element considered by itself, it does have responsibility for all system elements that constitute its system-of-interest and consequently for the output of projects at all levels subordinate to it, as shown in Figure 4.

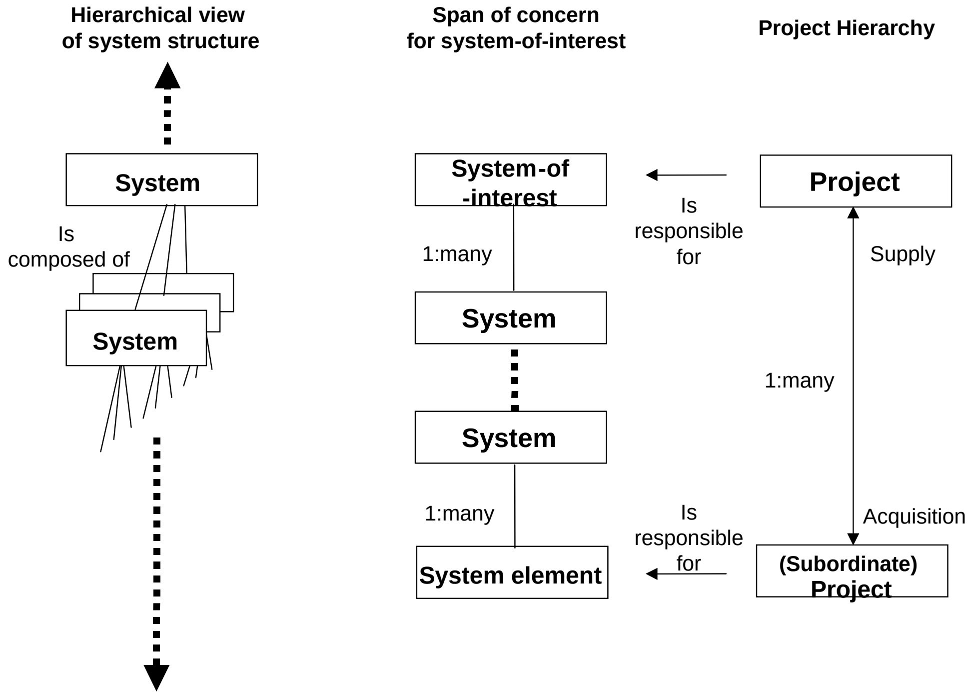

Figure 4—System and project hierarchies

In practice, the risks associated with implementing systems that fulfil specified requirements typically diminish with descending level of detail in the system-of-interest's structure and eventually are no longer of direct attention or concern to the particular project. At this level (not necessarily the same level down different paths of system-of-interest decomposition) a system element can be acquired with acceptable risk and the detail of its composition can remain hidden below some level. For example, if the system-of-interest for the project were a radar, in normal practice, the formulation and cure of the slurry mix for the resistive part of a composition resistor on a printed circuit board in a subassembly in a system element of the system-of-interest would not be addressed directly. Rather, the requirements at the system-of-interest and possibly the system element level would drive choices that would force a particular selection of this material in order to meet all the requirements for the higher level elements. From the system-of-interest view, the system elements may appear to be where specialist disciplines or particular implementation technology practices are present.

3.1.5 Enabling systems

Throughout the life cycle of a system-of-interest, essential services are required from systems that are not directly a part of the operational environment, e.g. mass-production system, training system, maintenance system. Each of these systems enables a part, e.g. a stage, of the life cycle of the system-of-interest to be conducted. Termed enabling systems, they facilitate progression of the system-of-interest through its life cycle.

The relationship between the services delivered to the operational environment by the system-of-interest and the services delivered by the enabling systems to the system-of-interest are shown in Figure 5. Enabling systems can be seen to contribute indirectly to the services provided by the system-of-interest.

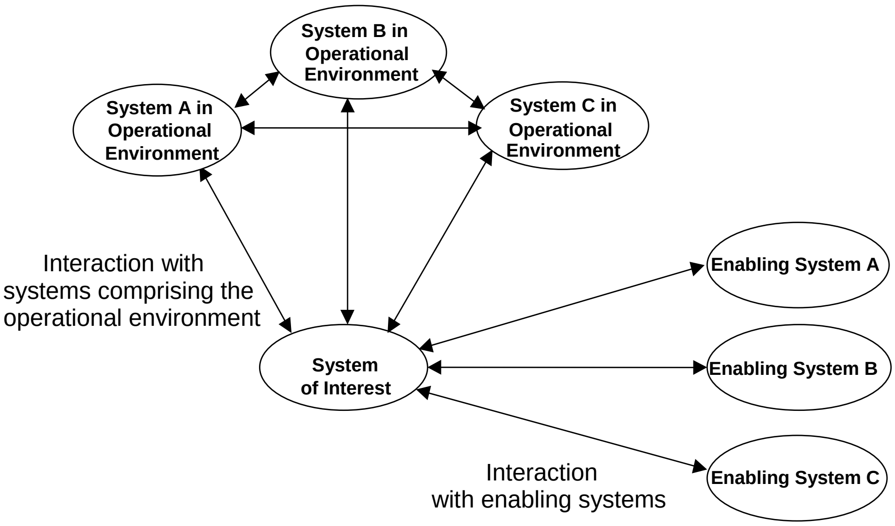

Figure 5—System-of-interest, its operational environment and enabling systems

During any particular stage in the system life cycle, the relevant enabling systems and the system-of-interest are considered together. Since they are interdependent, they can also be viewed as a system. Project responsibility for a stage in the life cycle of the system-of-interest thus extends to responsibility for acquiring services from the relevant enabling systems. When a suitable enabling system does not already exist, the project that is responsible for the system-of-interest can also be directly responsible for creating and using the enabling system.

3.2 Life cycle concepts

3.2.1 System life cycle model

Every system, whatever the kind or size, inherently follows some life cycle, evolving from its initial conceptualization through its eventual retirement, as defined in clause 2.14. It is generally useful to build a model of this progression to help manage the evolution of the system because movement from one stage to another represents a decision point with specific criteria to be satisfied before movement to the next stage is allowed. A life cycle model, then, is a decision-linked conceptual segmentation of the definition of the need for the system, its realization as a product or service, and its utilization, evolution and disposal. Note that the actual progression of a system through the parts of the model, however done, is the system's life cycle.

A system life cycle model is typically segmented by stages to facilitate planning, provisioning, operating and supporting the system-of-interest. These segments provide an orderly progression of a system through established decision-making gates to reduce risk and to ensure satisfactory progress. As stated before, it is the need to make a decision to specific criteria before a system can progress to the next stage that is the most important reason for using a life cycle model. A secondary aspect is that using a life cycle model can help an organization ensure that it thinks of its work and its processes within a larger framework, which may have useful business overtones. See, for example, the discussion of a services life cycle model in clause 7.1.

Several factors make system life cycle planning, provisioning and operation difficult to manage. Economics and market forces, as well as novelty, complexity and operational stability affect the length of a system life cycle. Some systems have life cycles that are decades long (for example aircraft, satellites, ships) and some are very short (for example instruments and consumer electronics).

A typical system, however, progresses through a common series of stages where it is conceptualized, developed, produced as a product or service, utilized, supported and retired. The life cycle model is the framework that helps ensure that the system is able to meet its required functionality throughout its life. Thus, to define system requirements and developing system solutions during the concept and development stages, experts from other stages (for example production, utilization, support, retirement) are needed to perform trade-off analyses and to help make design decisions and arrive at a balanced solution. This helps ensure that a system has the necessary attributes designed in as early as possible. Also, it is essential to have the necessary enabling systems available to perform required stage functions.

A representative system life cycle model, shown in Figure 6, illustrates this progression.

Concept	**Development**	**Production**	**Utilization**	**Support**	**Retirement**

Figure 6—Representative life cycle model

The stages in Figure 6, although drawn as discrete, are in practice interdependent and overlapping. Further, the figure implicitly conveys a uniformity and single linearity of time progression that is not inherently part of a life cycle model: stages do not necessarily occur one after another in time sequence. So, one actuality of the "progression" of a system through its life cycle could be as represented in Figure 7. When, in this Technical Report, reference is made to the next, subsequent, or later stage, this type of model must be kept in mind to avoid confusion by inferring linear time sequencing.

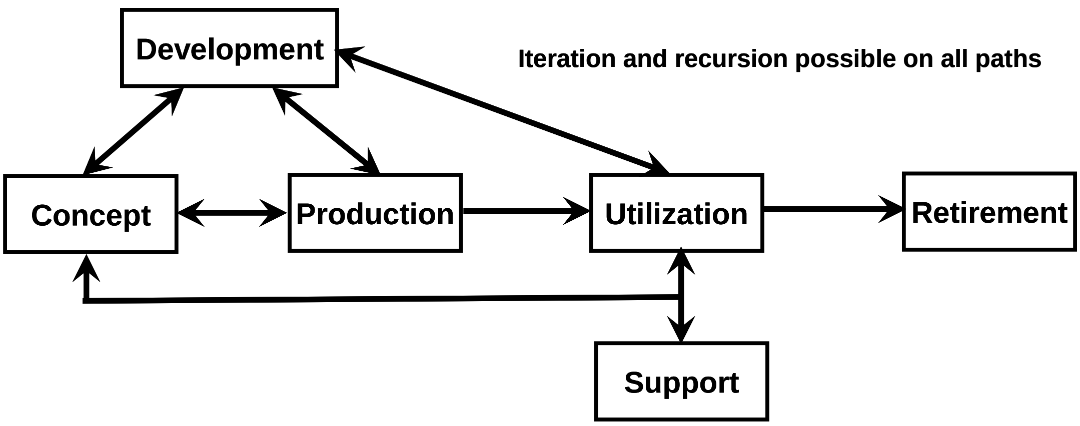

Figure 7—Life cycle model with some of the possible progressions

A system progresses through its life cycle as the result of actions, performed and managed by people in organizations, using processes for their performance. The detail in the life cycle model is expressed in terms of these processes, their outcomes, relationships and occurrence. A key aspect of life cycle management is that each stage requires exit criteria that must be met before progress from that stage is allowed. Since stages can have complex relationships as illustrated in Figure 7, definition of both exit and entry criteria for a specific stage progression may be needed. Thus, the use of stages corresponds to satisfying decision criteria before allowing further progress, providing a strong project control feature, whatever the sequence of stages within a given project's life cycle model.

Four common principles associated with a life cycle model are:

a) A system progresses through specific stages during its life;

b) Enabling systems should be available for each stage in order to achieve the outcomes of the stage;

c) At specific life cycle stages, attributes such as producibility, usability, supportability and disposability should be specified and designed into a system;

d) Progression to another stage requires satisfaction of exit criteria of the current stage.

The system life cycle model that is presented in Figure 6 is discussed in Clause 7 of ISO/IEC TR 19760 from an organizational view and an engineering view. The organizational view illustrates sequential, incremental, and evolutionary approaches. Another discussion of life cycle models is in Annex C of ISO/IEC TR 15271, where software is treated as a system-of-interest having its own life cycle, with three illustrative models being used for the discussion of the features, risks and opportunities of each model. Indeed, the Annex actually focuses on software development, thereby considering software development as a system-of-interest with its own life cycle.

3.2.2 System life cycle stages

Life cycles vary according to the nature, purpose, use and prevailing circumstances of the system. Nevertheless, despite an apparently limitless variety in system life cycles, there is an underlying, essential set of characteristic life cycle stages that exists in the complete life cycle of any system. Each stage has a distinct purpose and contribution to the whole life cycle and is considered when planning and executing the system life cycle.

The stages represent the major life cycle periods associated with a system and they relate to the state of the system description or realization of the system's set of products or services. The stages describe the major progress and achievement milestones of the system through its life cycle. They give rise to the primary decision gates of the life cycle. These decision gates are used by organizations to contain the inherent uncertainties and risks associated with costs, schedule and functionality when creating or utilizing a system. The stages thus provide organizations with a framework within which management has high-level visibility and control of project and technical processes.

Table 1 shows a commonly encountered example of life cycle stages. Also shown are the principal purposes of each of these stages and the possible decision options used to manage the achievement and risk associated with progression through the life cycle.

LIFE CYCLE STAGES	PURPOSE	DECISION GATES
CONCEPT	Identify stakeholders' needs Explore concepts Propose viable solutions	Decision Options: - Execute next stage - Continue this stage - Go to a preceding stage - Hold project activity - Terminate project
DEVELOPMENT	Refine system requirements Create solution description Build system Verify and validate system	
PRODUCTION	Produce systems Inspect and test	
UTILIZATION	Operate system to satisfy users' needs	
SUPPORT	Provide sustained system capability	
RETIREMENT	Store, archive or dispose of system	

Table 1—An example of stages, their purposes and major decision gates

Organizations employ stages differently to satisfy contrasting business and risk mitigation strategies. Using stages concurrently and, in a few cases, even in different orders, can lead to life cycle forms with distinctly different characteristics. Sequential, incremental or evolutionary life cycle forms are frequently used. Alternatively, a suitable hybrid of these can be developed. The selection and development of such life cycle forms by an organization depend on several factors, including the business context, the nature and complexity of the system, the stability of requirements, the technology opportunities, the need for different system capabilities at different times and the availability of budget and resources. In addition, major decision gates, often called milestones, and reviews to focus on making the decisions, may be incorporated by an organization on an incremental basis within a stage, as well as at the end of a stage, to further manage risks.

Just as all the system elements of a system contribute to the system as a whole, so each stage of the life cycle needs to be considered during any other stage of the life cycle. As a consequence, the contributing parties need to co-ordinate and co-operate with each other throughout the life cycle. This synergism of the life cycle stages and the functional contributors is necessary for successful project actions. Close communication with project team members from the different functions and organizations responsible for other life cycle stages leads to consistency in the life cycle.

3.2.3 Stages in a system-of-interest and its enabling systems

As with any system, each enabling system also has its own life cycle. Each life cycle is linked and synchronized to that of the system-of-interest. For example, if an enabling system does not already exist, its requirement is defined during the Concept Stage of the system-of-interest (or later if lead times permit), before the enabling system is utilized as shown in Figure 8 to provide its particular service to the system-of-interest.

An enabling system can pre-exist the system-of-interest, i.e. be an existing part of the infrastructure of the organization responsible for the system-of-interest or be in a service supplier's organization. Pre-existing enabling systems can introduce additional constraints on the system-of-interest.

Each enabling system can itself be considered as a system-of-interest, having in turn its own enabling systems. Therefore, the concepts in this Technical Report can also be applied to enabling systems.

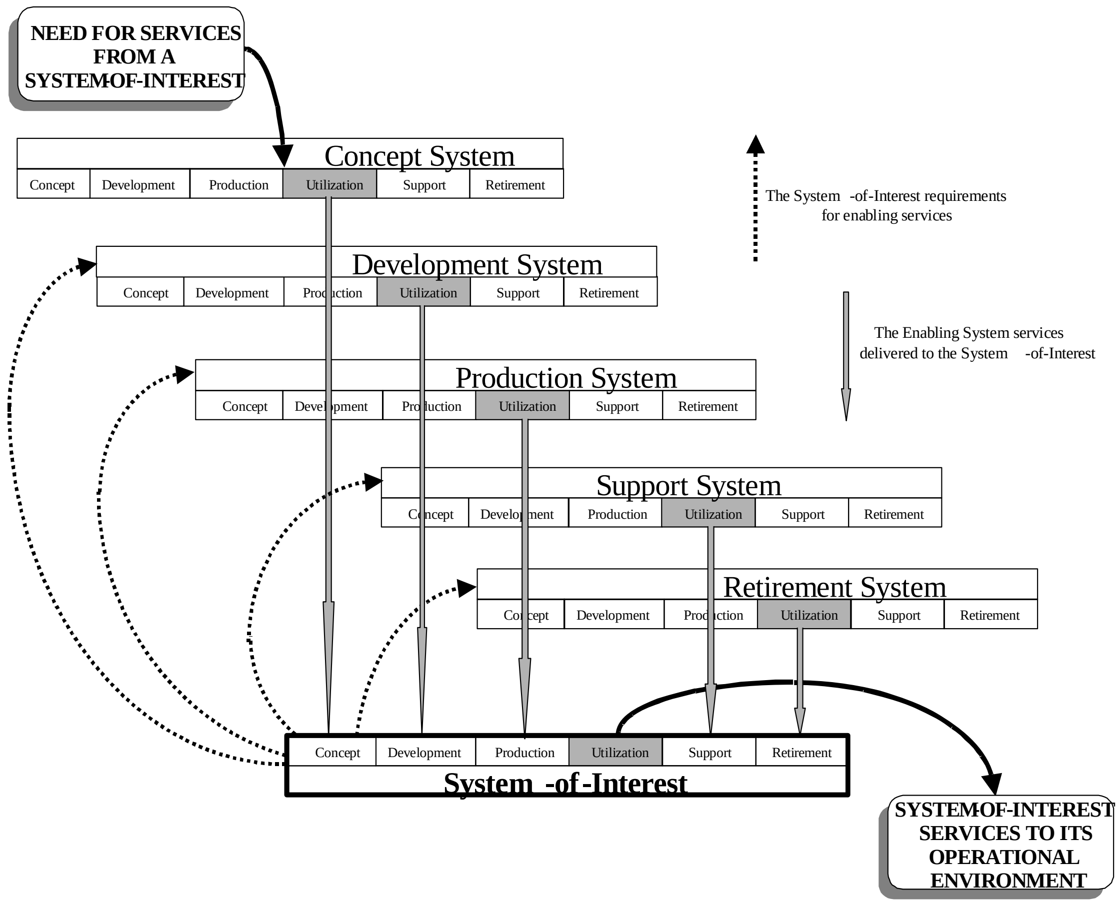

Figure 8—System interaction with typical enabling systems

3.3 Process concepts

3.3.1 Life cycle processes

The life cycle processes defined in this Technical Report can be used by any organization when acquiring and using, as well as when creating and supplying, a system. They can be applied at any level in a system's structure and at any stage in the life cycle.

The life cycle processes are based on principles of ownership (a process is associated with a responsibility, discussed further in Clause 3.3.2) and modularity. That is, the processes are:

a) Strongly cohesive, meaning that all the parts of a process are strongly related;

b) Loosely coupled, meaning that the number of interfaces among the processes is kept to a minimum.

The processes described in this Technical Report are not intended to preclude or discourage the use of additional processes that organizations find useful.

3.3.1.1 General

Within a life cycle stage, processes are performed as required to achieve stated objectives. The progression of a system through its life is the result of actions managed and performed by people in one or more organizations using the processes selected for a life cycle stage.

International Standard ISO/IEC 15288 is used to provide a specific example of four groups of system life cycle processes – agreement, organizational project-enabling, project and technical. Each process has a specific purpose, a set of expected outcomes and a set of activities. Each group of processes is described in Clause 5.3.2 of ISO/IEC 15288 and summarized in the following clauses of this Technical Report:

Guidance for the application of the International Standard ISO/IEC 15288 life cycle processes is provided in Clauses 6, 7 and 8 of ISO/IEC TR 19760. Similarly, guidance for ISO/IEC 12207 life cycle processes is provided in Clauses 6, 7 and 8 of ISO/IEC TR 15271.

Organizations, when considering a new project, should select a life cycle model, such as shown in Figure 6, and the necessary life cycle processes to satisfy applicable life cycle stage entry or exit criteria. Decisions as to which processes to select should be based on cost-benefit or risk reduction. The example process groups are described below.

3.3.1.2 Agreement processes

The agreement processes are applicable for establishing the relationship and requirements between an acquirer and supplier. The agreement processes provide the basis for initiation of other project processes to enable arriving at an agreement to engineer, utilize, support or retire a system and to acquire or supply related services.

The agreement processes can be used for several purposes, such as listed below, to:

a) Form and ensure completion of an agreement between an acquirer and a supplier for work on a system at any level of the system structure;

b) Establish and carry out agreements to acquire a system or related enabling system services;

c) Obtain work efforts by consultants, subcontractors, organizations, projects or individuals or teams within a project.

d) Provide the basis for closing an agreement after the system has been delivered or work has completed and payment made.

3.3.1.3 Organizational project-enabling processes

Organizational project-enabling processes are for that part of the general management that is responsible for establishing and implementing projects related to the products and services of an organization. Thus, the organization through these distinct processes provides the services that both constrain and enable the projects, directly or indirectly, to meet their requirements.

The organizational project-enabling processes included in International Standards such as ISO/IEC 12207 and ISO/IEC 15288 are not necessarily the only processes used by an organization for governance of its business. For example organizations also have processes for managing accounts receivable, accounts payable, payroll processing and marketing. These business-related processes are not directly within the scope of the mentioned standards and thus are not discussed further in this Technical Report. The Organizational project-enabling processes of ISO/IEC 15288 and ISO/IEC 12207 are constrained in their scope to the aspects of those processes that are required to bound and guide the project, even though there are implicit relationships that the organization must address elsewhere, for example, in the Human Resources Management Process.

For multiple projects involved in or interfacing with an organization, or for a teaming arrangement among external organizations, other organizational project-enabling processes can be appropriately tailored.

To perform these processes, it is not intended that a new organizational unit or discipline within an organization be created. Identified and defined roles, responsibilities and authorities may be assigned to individuals or established organizational units. When necessary, however, a new organizational unit can be formed.

The organizational project-enabling processes have specific objectives to fulfil such as listed below.

a) Provide the proper environment so that projects within the organization can accomplish their purpose and objectives.

b) Ensure that there is an orderly approach to starting, stopping and redirecting projects.

c) Ensure that organizational policies and procedures are defined that set forth the relevant life-cycle processes of the International Standards cited above and that are applicable to projects within the organization and its constituent parts.

d) Ensure that appropriate models, methods and tools are selected and provided to projects so that they can complete process activities efficiently and effectively.

e) Ensure that projects have adequate resources for the project to meet budget, schedule and performance requirements within acceptable risks and that human resources are appropriately trained for completing their responsibilities.

f) Ensure that project work products for delivery to customers are of a suitable quality.

3.3.1.4 Project processes

The project processes are used to manage technical process activities and to assure satisfaction of an agreement. Project processes are performed to establish and update plans, to assess progress against plans and system requirements, to control work efforts, to make required decisions, to manage risks and configurations and to capture, store, and disseminate information. Outcomes from performing the project processes help in the accomplishment of the technical processes.

The project processes apply to technology exploration projects that are most often part of larger projects. When that is the case, the appropriate project processes are performed at each level of the system structure. These processes also apply when performing organizational project-enabling processes or carrying out the activities related to a life cycle stage, including utilization, support and retirement.

When several projects co-exist within one organization, project processes should be defined to allow for the management of the resources and performance of the multiple projects.

3.3.1.5 Technical processes

The technical processes are applicable across all life cycle stages. For example, the following technical processes from ISO/IEC 15288 should be performed to engineer a system (a corresponding set for software is in ISO/IEC 12207):

a) Stakeholder Requirements Definition Process;

b) Requirements Analysis Process;

c) Architectural Design Process;

d) Implementation Process;

e) Integration Process;

f) Verification Process;

g) Transition Process;

h) Validation Process.

These processes should be performed to satisfy the entry or exit criteria of a system life cycle stage or set of stages. For example, they may be used during early system life cycle stages to create a feasible system concept, determine technology needs and establish future developmental costs, schedules and risks. During mid-system life cycle stages the technical processes may be used to define and realize a new system. During later system life cycle stages they may be used on legacy systems to make technology refreshments or technology insertions, as well as to correct variations from expected performance during production, utilization, support or retirement.

The other three technical processes (Operations Process, Maintenance Process and Disposal Process) can be used during any system life cycle to accomplish the objectives of a life cycle stage and support the technical processes used for engineering a system. The Operations Process and the Maintenance Process can be performed, as applicable, to support a particular version of a system. The Disposal Process can be performed to deactivate legacy systems, to dispose of legacy systems and to safely dispose of by-products from system use.

There may be additional technical processes used for a specific technical domain within a system, such as software, hardware, humans, or facilities, which are discussed in the appropriate International Standard for that technical domain (e.g. ISO/IEC 12207 for software).

3.3.2 Process responsibility

An organization is a body of entities organized for some specific purpose, and may be as diverse as a corporation, agency, society, union or club. An organization may be part of a parent organization (e.g, a society being parent to clubs that are a part of it). When an organization enters into an agreement, it is a party. Parties may have the same parent organization, although it is equally possible that they have different parent organizations (e.g. two clubs from the same society making an agreement, or two clubs from two different societies making an agreement).

Typically, organizations distinguish different areas of managerial responsibility and action through agreements, as indicated in Figure 9. Together, these areas contribute to the organization's overall capability to trade. This Technical Report employs a process model based on three primary organizational areas (or levels) of responsibility: organization, project and technical. Within each organization, a co-ordinated set of organizational project-enabling, project and technical processes contribute to the effective creation and use of systems, and therefore to achieving the organization's goals.

Each process is considered to be the responsibility of a party. An organization may perform one or more processes. A process may be performed by one organization or more than one organization, with one of the organizations being identified as the responsible party. A party executing a process has the responsibility for that entire process even though the execution of individual tasks may be by different parties. The responsibility aspect of the life cycle architecture facilitates adaptation and application of International Standards such as ISO/IEC 15288 or 12207 on a project, in which many parties may be legitimately involved.

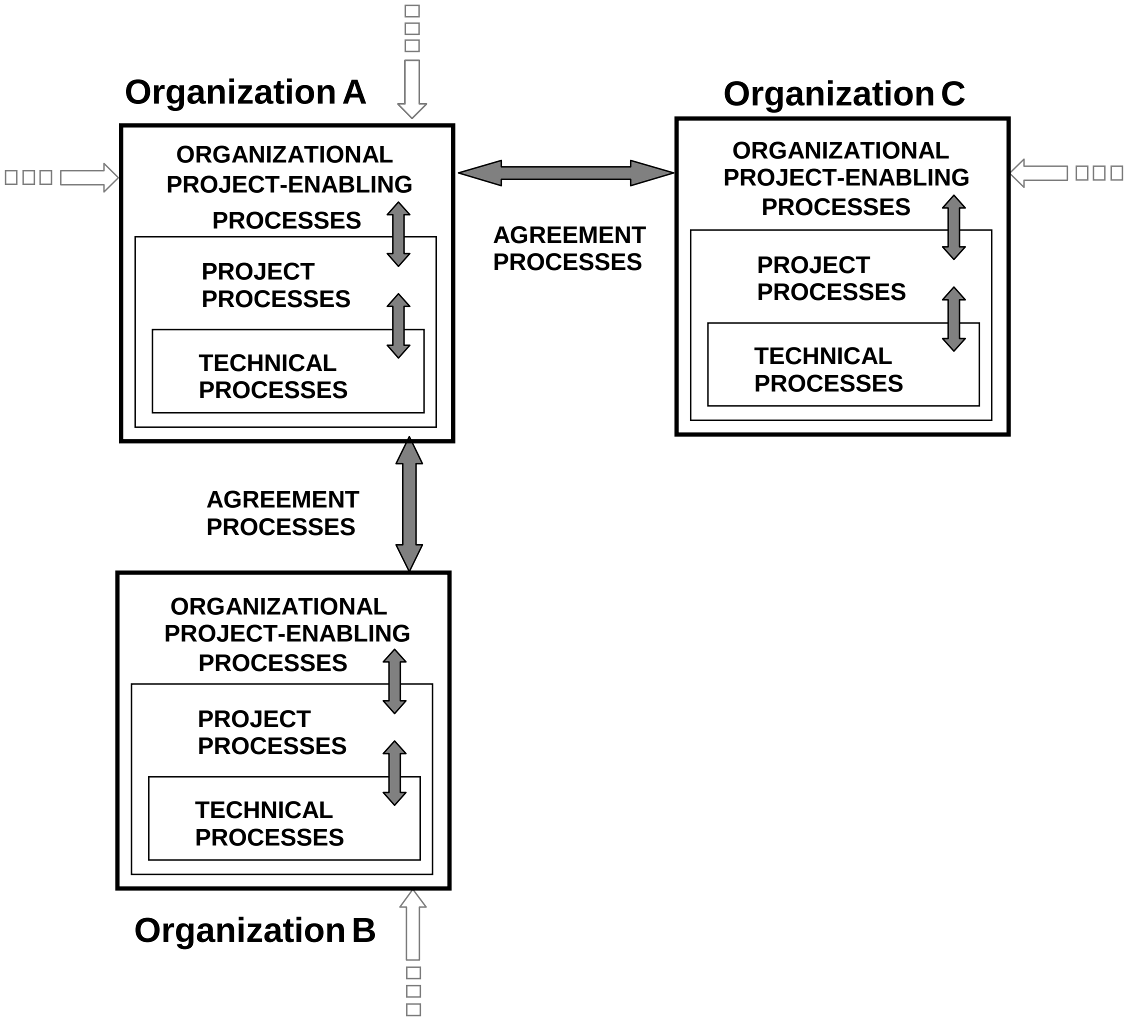

Figure 9—Agreements on responsibilities for organizational project-enabling, project and technical processes among cooperating organizations

Different organizations (or parts thereof) as parties, and different areas of responsibility within an organization, mutually establish their working relationships and acknowledge their respective responsibilities by making agreements. These agreements unify and co-ordinate the contributions made by different parties in order that they can meet a common business purpose.

3.4 Process application

Each system life cycle process in Figure 10 can be invoked, as required, at any point throughout the life cycle and there is no definitive order or time sequence in their use. The detailed purpose and timing of use of these processes throughout the life cycle are influenced by multiple factors, including social, trading, organizational and technical considerations, each of which can vary during the life of a system. An individual system life cycle is thus a complex system of processes that will normally possess concurrent, iterative, recursive and time dependent characteristics. Note that, in the figure, the clause numbers refer to the clauses in ISO/IEC 15288 where the processes are described, not to the clauses in this Technical Report. Note also that there is a corresponding figure (not shown) for software life cycle processes in ISO/IEC 12207, as there would be in analogous standards for other domains.

Concurrent use of processes can exist within a project, e.g. when design actions and preparatory actions for building a system are performed at the same time, and between projects, e.g. when system elements are designed at the same time under different project responsibility.

The iterative use of processes, i.e. the repeated application of a process or set of processes at the same hierarchical level of structural detail, is important for the progressive refinement of process outputs, e.g. the interaction between successive verification actions and integration actions can incrementally build confidence in the conformance of the product.

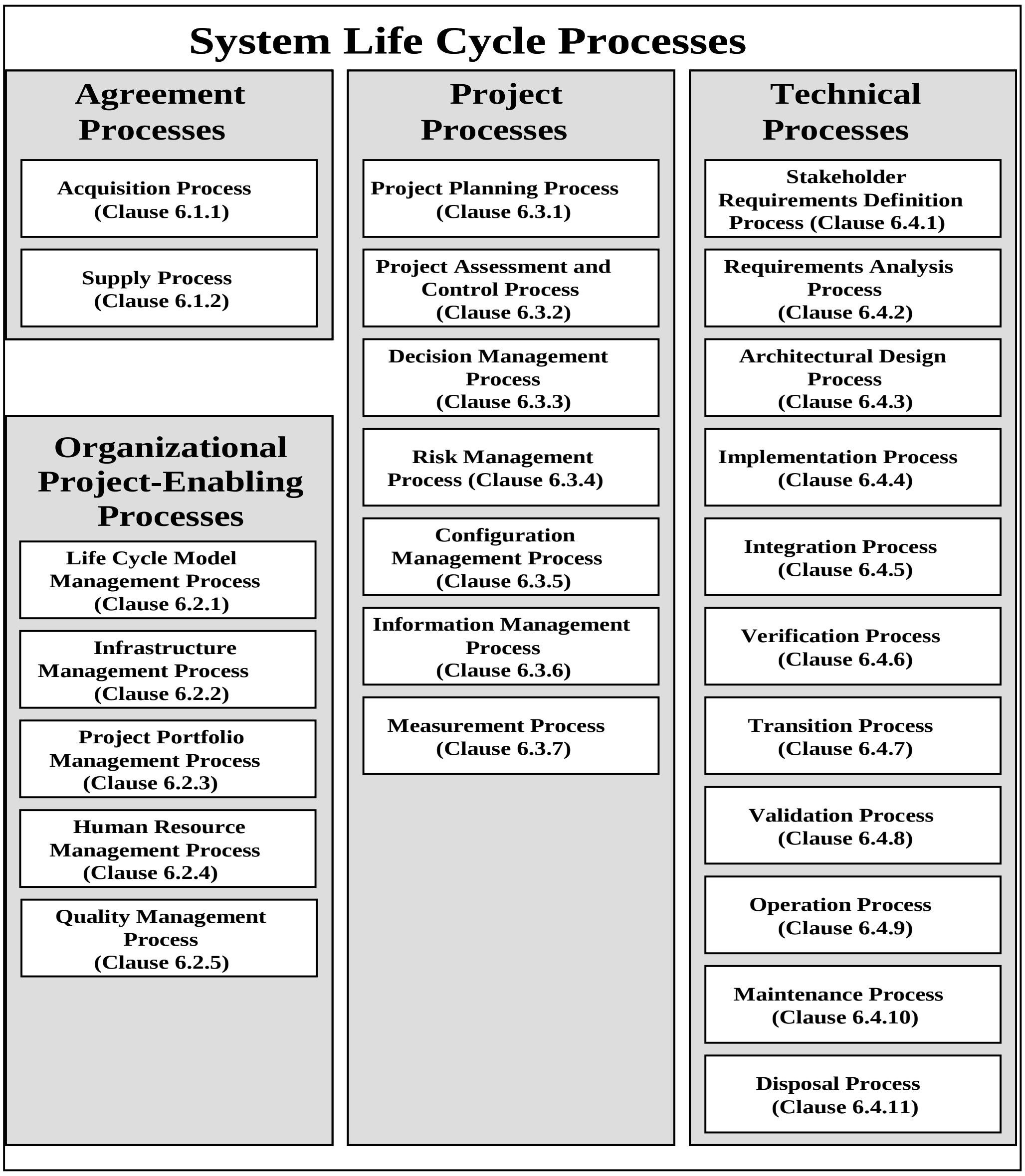

Figure 10—The system life cycle processes

The recursive use of processes, i.e. the repeated application of the same process or set of processes applied to successive levels of detail in a system's hierarchical structure, is a key aspect of the application of this Technical Report. The outputs of processes at any level, whether information, artefacts or services, are inputs to the same processes used at the level above or level below. This results in a response, information, artefacts or service, which can then modify the original output. In this way, the outputs across all levels of the system can be resolved and consistency achieved, e.g. system element descriptions that conform to an architecture.

The changing nature and complexity of the influences on the system (e.g. operational environment changes, new opportunities for system element implementation, modified structure and responsibilities in organizations) requires continual review of the selection and timing of process use. Process use in the life cycle is thus dynamic, responding to the many external influences on the system.

The life cycle stages assist the planning, execution and management of life cycle processes in the face of this complexity in life cycles by providing comprehensible and recognizable high-level purpose and structure. Precedence, particularly in similar market and product sectors, can assist the selection of stages and the application of life cycle processes to build an appropriate and effective life cycle model for any system.

3.5 Processes under key views

This Technical Report contains processes that are applicable throughout a life cycle. However, these processes may be used in different ways by different organizations and parties with different views and objectives. This clause presents processes and their relationships under key views. See clause 3.3.1 for synopses of the processes.

Figure 11 depicts examples of the life cycle processes and their relationships under different views of the usage of the Technical Report. The basic views shown are: contract, management, operating, engineering, and project.

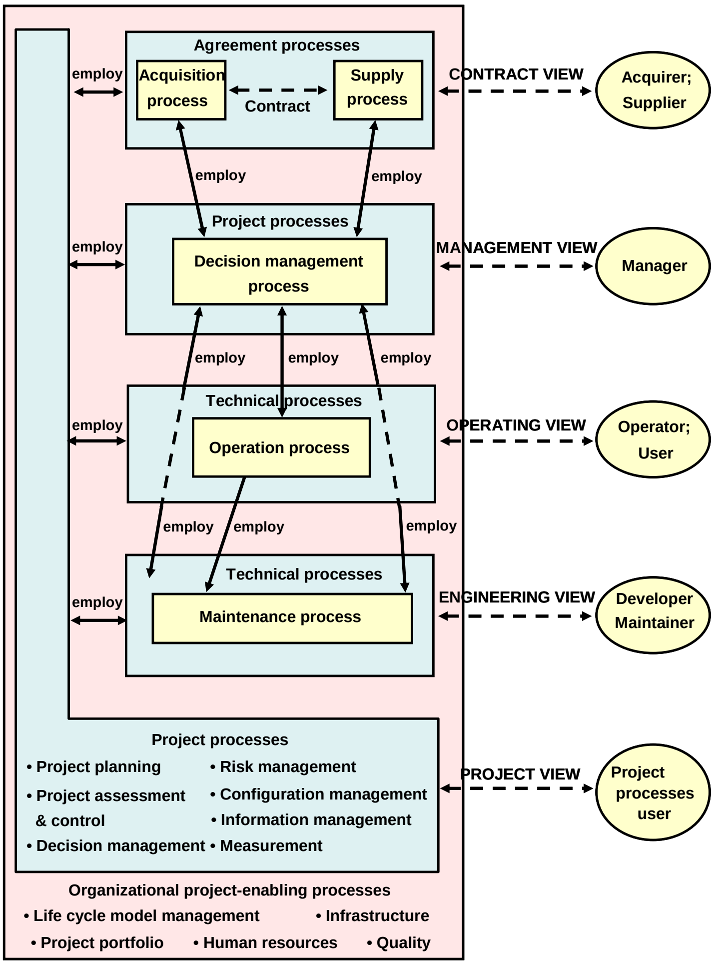

Figure 11—Life cycle processes roles and relationships

Under the contract view, acquirer and supplier parties negotiate and enter into a contract or other agreement and employ the Acquisition Process and Supply Process respectively. Under the management view, the acquirer, supplier, developer, operator, maintainer, or other party manages its respective process. Under the operating view, the operator provides operation service for the users. Under the engineering view, the developer or maintainer conducts its respective engineering tasks to produce or modify products or services. Under the project view, parties (such as configuration management or information management) provide supporting services to others in fulfilling specific, unique tasks. Also shown (see the bottom box) are some of the organizational project-enabling processes provided by the organization to support multiple projects; these are employed by an organization at a higher, e.g. corporate, level to establish and implement an underlying structure made up of associated life cycle processes and personnel and their continuous improvement.

The processes and parties (or stakeholders) are only related functionally. They do not dictate a structure for a party.

An organization or a party gets its name from the process it performs; for example, it is called an acquirer when it performs the Acquisition Process.

An organization may perform one process or more than one process; a process may be performed by one organization or more than one organization. Under one contract or application of this Technical Report, a given party should not perform both the Acquisition Process and the Supply Process, but it can perform other processes.

In the Technical Report itself, the relationships between the processes are only static. The more important dynamic, real-life relationships between the processes, between the parties, and between the processes and the parties are established when the appropriate International Standard (e.g. ISO/IEC 15288 for systems or 12207 for software) is applied on projects in a manner specific to that project. Each process (and the party performing it) contributes to the project in its own unique way. The Acquisition Process (and the acquirer) contributes by defining the product or service (or both) to be obtained. The Supply Process (and the supplier) contributes by providing the product or service. The Integration Process (and the integrator) contributes by "looking" to the system for correct derivation and definition of software, hardware and other products, as well as services, by supporting proper integration of products and services back into an overall system. The Operation Process (and the operator) contributes by operating the products or providing the services in the system's environment for the benefit of the users, the business, and the mission. The Maintenance Process (and the maintainer) contributes by maintaining and sustaining the products and services for operational fitness and by providing support and advice to the user community. Each process contributes by providing unique, specialized functions to other processes as needed.

Where processes are applied to system elements of a specific nature (e.g. software), the relevant processes from the standard that applies to that kind of element will be invoked (in this case, ISO/IEC 12207) as well as appropriate system processes. This holds true for elements of all types. Thus, there is a complementary duality of process interplay as the particulars of a system are instantiated. As a specific example, the validation process for software will be invoked as part of the overall process of validating the system, to ensure that the software is validated as part of the system.

4 Life cycle stages

4.1 Introduction

Life cycle stages are the specific framework within which system life cycle processes are applied to life cycle models. The value of stages is that entry into and exit from each stage represents a decision point. Therefore, progress is explicitly gated and the criteria for each step of progress-moving to the next stage—is also explicit. These criteria determine when a stage can be exited and when the next appropriate stage can be commenced. For stages that follow each other in linear fashion, the exit and entrance criteria are the same. However, more complex stage relationships may necessitate separate exit and entrance criteria.

Each life cycle process of ISO/IEC 12207 and ISO/IEC 15288 can be invoked at any point throughout the life cycle, as was noted in Clauses 3.3 and 3.4. The order and sequence in which processes are invoked, and when they are invoked, is driven by the project requirements and context: there is no unique definitive order or time sequence for their use within a stage or across several stages.

So, system, software, service (or other) life cycle processes can be invoked concurrently, iteratively, recursively and time dependently in whatever stages of an individual system, software, service, or other, life cycle model are applicable to a given project. The scale and rigour of process application in the listed stages and the duration of these stages will be determined by the varying technical and business needs of the projects defining and using the life cycle model.

The processes invoked are generally performed by systems other than the system-of-interest, e.g. by enabling systems.

This Technical Report describes the following six stages as examples:

a) Concept Stage;

b) Development Stage;

c) Production Stage;

d) Utilization Stage;

e) Support Stage;

f) Retirement Stage.

In the following clauses of this Technical Report, for each example stage an overview description is given, followed by descriptions of the example stages purpose and outcomes.

4.2 Concept stage

4.2.1 Overview

The Concept Stage begins with initial recognition of a need or a requirement for a new system-of-interest or for the modification to an existing system-of-interest. This is an initial exploration, fact finding, and planning period, when economic, technical, strategic, and market bases are assessed through acquirer/market survey, feasibility analysis and trade-off studies. Acquirer/user feedback to the concept is obtained.

One or more alternative concepts to meet the identified need or requirement are developed through analysis, feasibility evaluations, estimations (such as cost, schedule, market intelligence and logistics), trade-off studies, and experimental or prototype development and demonstration. The need for one or more enabling systems for development, production, utilization, support and retirement of the system-of-interest is identified and candidate solutions are included in the evaluation of alternatives in order to arrive at a balanced, life cycle solution. Typical outputs are stakeholder requirements, concepts of operation, assessment of feasibility, preliminary system requirements, outline design solutions in the form of drawings, models, prototypes, etc., and concept plans for enabling systems, including whole life cost and human resource requirements estimates and preliminary project schedules. Decisions are made whether to continue with the implementation of a solution in the Development Stage or to cancel further work.

It is presumed that the organization has available enabling systems for the concept stage that consist of the methods, techniques, tools and competent human resources to undertake market/economic analysis and forecasting, feasibility analysis, trade-off analysis, technical analysis, whole life cost estimation, modelling, simulation, and prototyping.

4.2.2 Purpose

The Concept Stage is executed to assess new business opportunities and to develop preliminary system requirements and a feasible design solution.

4.2.3 Outcomes

The outcomes of the Concept Stage are listed below:

a) The identification of new concepts that offer such things as new capabilities, enhanced overall performance, or reduced stakeholders' total ownership costs over the system life cycle.

b) An assessment of feasible system-of-interest concepts and solutions, including enabling systems throughout the life cycle, for closure against both technical and business stakeholder objectives.

c) The preparation and baselining of stakeholder requirements and preliminary system requirements (technical specifications for the selected system-of-interest and usability specifications for the envisaged human-machine interaction).

d) Refinement of the outcomes and cost estimates for stages of the system life cycle model.

e) Risk identification, assessment and mitigation plans for this and subsequent stages of system life cycle model.

f) Identification and initial specification of the services needed from enabling systems throughout the life of the system.

g) Concepts for execution of all succeeding stages.

h) Plans and exit criteria for the Development Stage.

i) Development and satisfaction of stage exit criteria.

j) Approval to proceed to the Development Stage.

4.3 Development stage

4.3.1 Overview

The Development Stage begins with sufficiently detailed technical refinement of the system requirements and the design solution and transforms these into one or more feasible products that enable a service during the Utilization Stage. The system-of-interest may be a prototype in this stage. The hardware, software and operator interfaces are specified, analyzed, designed, fabricated, integrated, tested and evaluated, as applicable, and the requirements for production, training, and support facilities are defined. This stage also ensures that the aspects of other stages (production, utilization, support, and retirement) and their enabling systems' requirements and capabilities are considered and incorporated into the design through the involvement of all interested parties. Feedback is obtained from stakeholders and those who will produce, operate, use, support, and retire the system-of-interest through such means as a series of technical or other reviews. Outputs are a system-of-interest or a prototype of the final system-of-interest, refined requirements for enabling systems or the enabling systems themselves and all documentation and cost estimates of future stages.

Planning for this stage begins in the preceding stage to ensure the organization has available, or can establish, an infrastructure of development enabling systems, consisting of facilities, processes, procedures, methods, techniques, tools and competent human resources to undertake analysis, modelling and simulation, prototyping, design, integration, test and documentation. These items are developed or acquired to be available when needed to support development.

4.3.2 Purpose

The Development Stage is executed to develop a system-of-interest that meets stakeholder requirements and can be produced, tested, evaluated, operated, supported and retired.

4.3.3 Outcomes

The outcomes of the Development Stage are listed below:

a) Baselined technical information, including evaluated and refined system requirements and:

- o Hardware diagrams, drawings and models;
- o Software design documentation;
- o Interface specifications;
- o Production plans;
- o Operating instructions;
- o Training manuals for operators;
- o Maintenance procedures;
- o Retirement considerations.

b) Project budget and schedule baselines and life cycle ownership cost estimates.

c) A system-of-interest structure comprised of, for example, hardware elements, software elements, human elements and the interfaces (internal and external) of all such elements.

d) Verification and validation documentation.

e) Evidence supporting a decision, with all risks and benefits considered, that the system-of-interest meets all specified requirements and is producible, operable, supportable and capable of retirement and is cost effective for stakeholders.

f) Refined and baselined requirements for the enabling systems.

g) A prototype or final system-of-interest.

h) Refined outcomes and cost estimates for the Production, Utilization, Support and Retirement Stages.

i) Definition of the enabling system services required in subsequent life cycle stages.

j) Plans and exit criteria for the Production Stage.

k) Identification of current risks and determination of their treatment.

l) Satisfaction of stage exit criteria.

m) Approval to proceed to the Production Stage.

4.4 Production stage

4.4.1 Overview

The Production Stage begins with the approval to produce the system-of-interest. The system-of-interest may be individually produced, assembled, integrated, and tested, as appropriate, or may be mass-produced. Planning for this stage begins in the preceding stage. Production may continue throughout the remainder of the system life cycle. During this stage, the system may undergo enhancements or redesigns, the enabling systems may need to be reconfigured and production staff re-trained in order to continue evolving a cost effective service from the stakeholder view.

It is presumed that the organization has available the budget and enabling systems that consist of production equipment, facilities, tools, processes, procedures and competent human resources. These items are developed or acquired in order to be available when needed to enable production.

4.4.2 Purpose

The Production Stage is executed to produce or manufacture the system-of-interest, test it and produce related supporting and enabling systems as needed.

4.4.3 Outcomes

The outcomes of the Production Stage are listed below:

a) Qualification of the production capability.

b) Acquisition of resources, material, services and system elements to support the target production quantity goals.

c) The system produced according to approved and qualified production information.

d) Packaged product transfer to distribution channels or acquirer.

e) Plans and exit criteria for the Utilization Stage and the Support Stage.

f) Updated concepts for execution of all succeeding stages.

g) Current risks and mitigating actions identified.

h) Quality assured systems-of-interest accepted by the acquirer.

i) Satisfaction of stage exit criteria.

j) Approval to proceed to the Utilization Stage.

k) Approval to proceed to the Support Stage.

4.5 Utilization stage

4.5.1 Overview

The Utilization Stage begins after installation and transition to use of the system. The Utilization Stage is executed to operate the product at the intended operational sites to deliver the required services with continuing operational and cost effectiveness. This stage ends when the system-of-interest is taken out of service.

Planning for this stage begins in the preceding stages. This stage includes those processes related to use of the system to provide services, as well as monitoring performance and identifying, classifying and reporting of anomalies, deficiencies, and failures. The response to identified problems includes taking no action; maintenance and minor (low cost/temporary) modification (reference Support Stage); major (permanent) modification and system-of-interest life extensions (reference Development and Production Stages), and end-of-life retirement (reference Retirement Stage).

During this stage the product or services can evolve giving rise to different configurations. The operator operates the different configurations and the responsible product supplier manages the status and descriptions of the various versions and configurations of the product or services in use.

It is presumed that the organization has available the utilization stage enabling system which consists of facilities, equipment, processes, procedures, trained personnel and instruction manuals. These items are developed or acquired in order to be available when needed to support utilization.

4.5.2 Purpose

The Utilization Stage is executed to operate the product, to deliver services within intended environments and to ensure continuing operational effectiveness.

4.5.3 Outcomes

The outcomes of the Utilization Stage are listed below:

a) Experienced personnel with the competence to be operators in the system-of-interest and provide operational services.

b) An installed system-of-interest that is capable of being operated and of providing sustainable operational services.

c) Performance and cost monitoring and assessment to confirm conformance to service objectives.

d) Identification of problems or deficiencies, informing appropriate parties (user, development, production, or support) of the need for corrective action.

e) New opportunities for system-of-interest enhancement through stakeholder feedback.

f) Plans and exit criteria for the Retirement stage.

g) Satisfaction of stage exit criteria.

h) Approval to proceed to the Retirement Stage.

4.6 Support stage

4.6.1 Overview

The Support Stage begins with the provision of maintenance, logistics and other support for the system-of-interest's operation and use. Planning for this stage begins in the preceding stages. The Support Stage is completed with the retirement of the system-of-interest and termination of support services.

This stage includes those processes related to providing services that support utilization of the system-of-interest. This stage also includes processes to use and monitor the support system itself and its services, including the identification, classification, and reporting of anomalies, deficiencies, and failures of the support system and services. Actions to be taken as a result of identified problems with the support system include maintenance and minor modification of the support system and services, major modification of the support system or services (reference Development and Production Stages), and end-of-life retirement of the support system and services (reference Retirement Stage).

During this stage the support system and services can evolve under different versions or configurations. The support organization operates the different versions or configurations and the responsible product organization manages the status and descriptions of the various versions and configurations of the support system and services in use.

It is presumed that the supporting organization has available the enabling systems, which consist of facilities, equipment, tools, processes, procedures, trained support personnel, and maintenance manuals. The items making up the support enabling system are developed and acquired in order to be ready when needed to support the system-of-interest.

4.6.2 Purpose

The Support Stage is executed to provide logistics, maintenance, and support services that enable continuing system-of-interest operation and a sustainable service.

4.6.3 Outcomes

The outcomes of the Support Stage are listed below:

a) Trained personnel who will maintain the system and provide the support services.

b) Organizational interfaces with the operating and production organizations that ensure problem resolution and corrective actions.

c) Maintained product and services and the provision of all related support services, including logistics, to the operational sites.

d) Product and service maintenance and corrected design deficiencies.

e) All required logistics support, including a spare parts inventory sufficient to satisfy operational availability goals.

f) Current risks and mitigating actions identified.

g) Agreement to terminate support services.

h) Satisfaction of stage exit criteria.

4.7 Retirement stage

4.7.1 Overview

The Retirement Stage provides for the removal of a system-of-interest and related operational and support services. Planning for the Retirement Stage begins in the preceding stages. This stage begins when a system-of-interest is taken out of service.

This stage includes those processes related to operating the system that enables retirement of the system-of-interest (the retirement enabling system) and also includes monitoring performance of that enabling system and the identification, classification, and reporting of anomalies, deficiencies, and failures of the retirement enabling system. Actions to be taken as a result of identified problems include maintenance and minor modification of the retirement enabling system (reference Support Stage), major modification of the retirement enabling system (reference Development and Production stages), and end-of-life retirement of the retirement enabling system itself (reference Retirement Stage).

It is presumed that the organization has access to an enabling system, which consists of facilities, tools, processes, procedures, equipment, trained personnel and, as appropriate, access to recycling, disposal or containment facilities. The items making up the retirement enabling system are developed and acquired in order to be ready when needed to perform retirement functions.

This stage is applicable whenever a system-of-interest reaches its end-of-service life. Such end-of-service life can be the result of replacement by a new system, irreparable wear, catastrophic failure, no further use to the user, or not cost effective to continue operating and supporting the system-of-interest.

4.7.2 Purpose

The Retirement Stage is executed to provide for the removal of a system-of-interest and related operational and support services, and to operate and support the retirement system itself.

4.7.3 Outcomes

The outcomes of the Retirement Stage are listed below:

a) Experienced personnel who can provide retirement services.

b) Required system-of-interest decommissioning, including disposal, refurbishing, or recycling, in accordance with applicable health, safety, security, privacy and environmental laws and regulations.

c) Plans and procedures for transferring the provision of services to the new system-of-interest if applicable.

d) Removal of waste.

e) Environment returned to original or agreed state.

f) Archived elements.

g) Operational staff is redeployed.

h) Satisfaction of stage exit criteria.

5 Life cycle model illustrations using ISO/IEC 15288 and 12207

This clause illustrates generic life cycle models of a system or project and describes how ISO/IEC 15288 and 12207 can be applied within those system life cycle models. In addition, the two life cycle models are compared.

5.1 System life cycle model using ISO/IEC 15288

A typical system life cycle model begins with the conception of an idea or a need, runs through the development, production, operation, and maintenance of the system, and ends with its retirement. A life cycle model is normally divided using one or more logical dividing bases, such as objective met, milestone, or increment of time. Each division includes major, distinct activities and tasks to be performed and may require an authorization at a transition. Stages are often, but not necessarily always, sequential. For example, a generic system life cycle model could be divided as follows, with each division being adaptable to a particular system life cycle model:

a) Concept Stage;

b) Development Stage;

c) Production Stage;

d) Utilization Stage;

e) Support Stage;

f) Retirement Stage.

5.1.1 Example of ISO/IEC 15288 in a generic system life cycle model

Figure 12 of this Technical Report presents highlights of the use of ISO/IEC 15288 in a generic example system life cycle model. Note that the model does not show the Organizational Project-enabling Processes, which are utilized as appropriate throughout the life cycle stages.

The Figure shows the stages as if they were sequential. However, additional arrows have been added to the side of each stage to convey the idea that one can jump from a stage to one that does not immediately follow it, or revert to a prior stage or stages that do not immediately precede it. Further, the text in the model indicates that one applies, at any stage, the appropriate life cycle processes, in whatever sequence is appropriate to the project, and repeatedly or recursively if appropriate. While this may seem to be a total lack of structure, indeed it is not. Rather the structure has well defined parts that can be juxtaposed as needed to get the job done, flexibly but still in a disciplined manner, just as a real structure would be created.

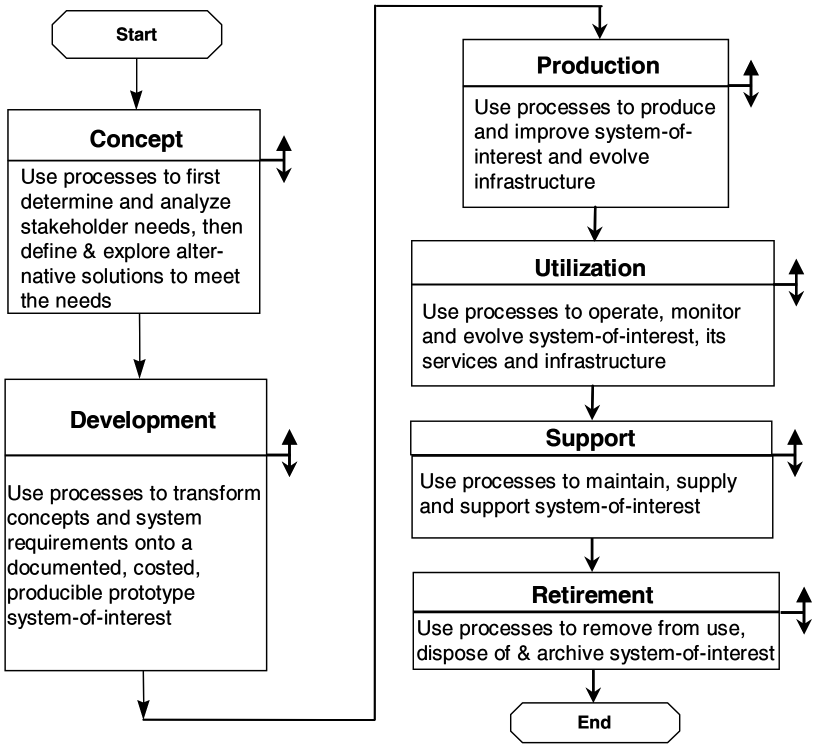

Figure 12—Using ISO/IEC 15288 to support a system life cycle model

In any stage or across the total life cycle model, an organization may use ISO/IEC 15288 internally or may retain a supplier to provide the products or services in whole or part.

5.1.1.1 Concept stage

During this stage, after initial recognition of a need, one or more alternative solutions to meet the identified need or concept are explored and developed through analysis, feasibility evaluations, estimations (such as cost, schedule, market intelligence and logistics), trade-off studies, and experimental or prototype development and demonstration. Acquirer/user feedback to the concept is obtained. A specific development strategy is selected, such as one of these described in Annex A.

5.1.1.2 Development stage

This stage begins with sufficiently detailed technical refinement of the system requirements and the design solution and transforms these into one or more feasible products that enable a service during the Utilization Stage. Planning for this stage begins in the preceding stage to ensure the organization has available, or can establish, an infrastructure to enable development. Outputs are a system-of-interest or a prototype of the final system-of-interest, refined requirements for enabling systems or the enabling systems themselves and all documentation and cost estimates of future stages.

5.1.1.3 Production stage

During this stage, with approvals, the system-of-interest may be individually produced, assembled, integrated, and tested, as appropriate, or may be mass-produced. The system-of-interest is installed and transitioned into use.

The product may undergo enhancements or redesigns, with our without user/operator/maintainer feedback. The enabling systems may need to be reconfigured, the production infrastructure may need to be modified and production staff re-trained in order to continue evolving a cost effective service from the stakeholder view.

5.1.1.4 Utilization stage

The Utilization Stage is executed to operate the product at the intended operational sites to deliver the required services with continuing operational and cost effectiveness. This stage includes those processes related to use of the product to provide services, as well as monitoring performance and identifying, classifying and reporting of anomalies, deficiencies, and failures. During this stage the product or services, as well as the operational infrastructure, can evolve giving rise to different configurations. This stage ends when the system-of-interest is taken out of service.

5.1.1.5 Support stage

The Support Stage provides maintenance, logistics and other support for the system-of-interest's operation and use. This stage includes those processes related to operating the support system and providing support services to users of the system-of-interest. This stage also includes monitoring performance of the support system and services and the identification, classification, and reporting of anomalies, deficiencies, and failures. During this stage the product or services, as well as the operational infrastructure, can evolve giving rise to different configurations. This stage ends when the system-of-interest is taken out of service.

5.1.1.6 Retirement stage

The Retirement Stage provides for the removal of a system-of-interest and related operational and support services from normal use. This stage includes those processes related to operating the retirement system and also includes monitoring performance of the retirement system and the identification, classification, and reporting of anomalies, deficiencies, and failures of the retirement system. This stage begins when a system-of-interest is taken out of service and ends when all elements have been properly disposed of and required archives established.

5.2 Software life cycle model using ISO/IEC 12207

A typical software life cycle model consists of several stages. It begins with an idea or a concept for a software product or service, continues through systems engineering and software engineering, and advances to operation, maintenance and support, and ends in retirement. ISO/IEC 12207 organizes these and related stages into the primary, supporting, and organizational processes that make up the software life cycle model.

5.2.1 Example of ISO/IEC 12207 in a generic software life cycle model

Figure 13 of this Technical Report presents highlights of the use of ISO/IEC 12207 in a generic example software life cycle model. Its basic purpose is stated briefly, followed by the mode of use of ISO/IEC 12207. Note that the model does not show the Organizational Project-enabling Processes, which are utilized as appropriate throughout the life cycle stages.

In any stage or across the total life cycle model, an organization may use ISO/IEC 12207 internally or may retain a supplier to provide the products or services in whole or part.

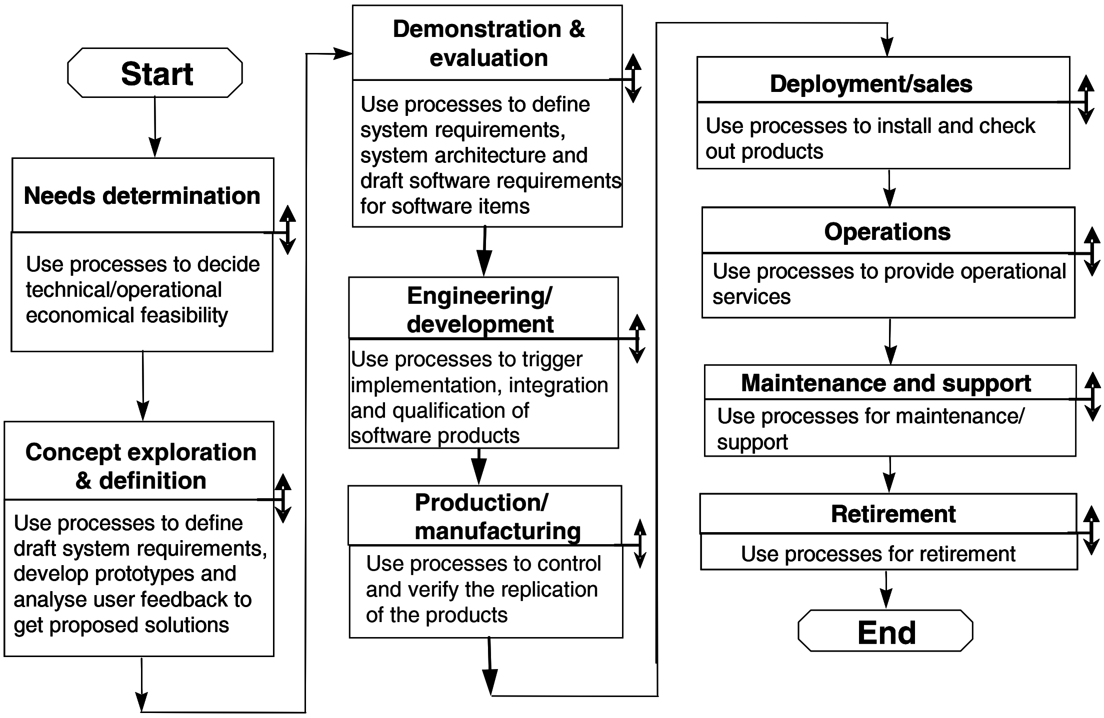

Figure 13—Using ISO/IEC 12207 to support a software life cycle model

5.2.1.1 Needs determination stage

During this stage, an idea or a need for a new or improved system is identified and determined. A top-level statement of need is developed, along with a review of such items as the cost, criticality, and feasibility of the intended system.

The Acquisition process may be used to help decide whether the need is technologically or operationally feasible before proceeding with further studies, development, and commitment. Further, selected Technical Processes may be used to develop software, or the methods or models employed in making the decisions. Possibly, a specific development strategy is selected, such as one of these described in Annex A, though this may be deferred to either of the following two stages, depending on the degree of exploration, prototyping and evaluation to be done.

5.2.1.2 Concept exploration and definition stage

This stage is the initial planning period when technical, strategic, and economic market bases are assessed through comprehensive studies, experimental development, and concept evaluation. Solutions proposed to meet an identified need may be refined, or alternative solutions may be developed through feasibility evaluations, estimates (such as cost, schedule, marketing, intelligence and logistics), trade-off studies, and analyses. Outputs of this stage are generally preliminary system requirements, and possibly software for prototypes, that feed into the next stage or stages.

The Acquisition, Supply, and selected Technical processes may be used to:

a) Help determine preliminary system requirements;

b) Develop prototypes;

c) Analyse and incorporate user's feedback to the proposed solutions.

The Technical processes may be used to develop software of the analytical/simulation models needed in studies and decision making and definition of solutions.

5.2.1.3 Demonstration and evaluation stage

During this stage, system characteristics, concepts, and solutions, including computer resources, are further defined through systems engineering, development of preliminary equipment and prototype software, and testing and evaluation. The system characteristics and preferred concepts and solutions are evaluated to demonstrate that the system, including its computer hardware and software resources, is suitable for engineering development. System requirements are baselined and are allocated initially to system elements, such as hardware, software, facilities, personnel, processes, procedures and naturally occurring entities. These outputs feed into the next stage or stages.

The Acquisition, Supply, and selected Technical Processes may be used for analysing and defining system requirements, top-level system design solutions, and preliminary requirements for system elements, including software. Other Technical Processes may be used as a basis for analysis, demonstration, validation, testing, prototyping of requirements and design solutions.

5.2.1.4 Engineering/development stage

This stage is the period when the system hardware, computers, software, facilities, personnel sub-systems, training, and support items are designed, fabricated, integrated, tested, and evaluated. The output is a system which closely approximates the production item, the documentation necessary for the next stage or stages, and the test results that qualify the system to be produced.

ISO/IEC 12207 is designed to be fully applicable during this stage. The sub-processes, activities, and tasks of the Acquisition, Supply, and Technical Processes should be selected, tailored appropriately, and applied to perform software development or upgrades. This stage may include one or more iterations of the Technical Processes executed in coordination with other system elements. Outputs are baselined software requirements, design, and code.

If the software to be developed is to be a part of the system, then all the activities of the Technical Processes should be required. In addition, it needs to be clarified whether the developer would perform or support the system-related activities. If the software to be developed is to be stand-alone and not a part of the system, then the system-related activities should not be required, but should be considered.

5.2.1.5 Production/manufacturing stage

During this stage, the engineered and developed system undergoes production for the acquirer (which may be the user and/or operator) or manufacturing for the market (e.g. the acquirer vends directly or indirectly to users who were not specifically pre-determined). The production period contains both an approval to produce and the actual production until the system is delivered and accepted. The objective is to efficiently produce and deliver the working and supported system to the acquirer. The manufacturing period is from the approval to manufacture and manufacturing until the system is redesigned or retired. The objective is to efficiently manufacture and deliver working and supported systems to the consumers.

For software, production/manufacturing is minimal when compared to hardware. It involves copying the developed software and documentation onto appropriate media for various users/consumers. (There are no explicit tasks for this in ISO/IEC 12207.) Local industry practices and government regulations may be applicable. The Configuration Management process may be used for the control of these related tasks. Other activities such as Quality Assurance may be used where appropriate.

5.2.1.6 Deployment/sales stage

During this stage, the system undergoes deployment for the acquirer or sales to consumers. The deployment period commences with delivery of the first operational system to the acquirer for use, operation and maintenance by users, operators and maintainers. The sales period is from the distribution of the first batch of systems to the consumers until the system is withdrawn from the market.

The Acquisition, Supply, and Installation, Acceptance Support and Verification processes may be used for installation and check-out of the developed or modified software.

5.2.1.7 Operations stage

This stage covers operation, execution, or use of the system by the users and consumers and ends when the system is removed from operations.

The Acquisition, Supply, and Operation processes may be used for the operation of the software and for providing operational support to the associated users.

5.2.1.8 Maintenance and support stage

During this stage the system is modified due to errors, deficiencies, problems, user requests, or the organizational need for adaptation or improvement. This stage includes provision of logistics, technical, and repair support to users (or consumers).

The Acquisition, Supply, and Maintenance processes may be used for the maintenance of the software and for providing support services to the organization, users, and consumers.

All interfaces with the Technical Processes need to be determined. Depending on the impact of the work, the activities from the Technical Processes which are needed may vary taking into account the applicable situation.

5.2.1.9 Retirement stage

During this period, the system is retired from normal services. It includes archiving the retiring system and providing limited support to its users for a given period.

The Acquisition process and the Disposal process may be used for retiring the software and for providing support services to the organization, users, and consumers for a specific period.

5.3 Adapting ISO/IEC 15288 and ISO/IEC 12207 life cycle models

The two life cycle models shown, systems in Figure 12 and software in Figure 13, are representative of how life cycle stages are partitioned for each of those areas. However, for any given situation, it may better suit the purposes of the organization to have a different partitioning of its life cycle model. The result is that a specific system project could unfold best using the life cycle model shown in Figure 12 and conversely, a software project might fare best applying the model shown in Figure 13. Yet a third project might find that neither model fits its situation, so a different model should be devised. The key point is that the life cycle model should be tailored or adapted as is discussed in clause 6, rather than a single example being blindly adhered to.

6 Life cycle adaptation

6.1 Introduction

No two projects are the same. Each organization is driven by the nature of its business, its social responsibilities, and its business strategy. These provide constraints on available business opportunities that the organization and its projects can exploit. To help exploit opportunities, the organization establishes policies and procedures to guide the performance of projects. Variations in organizational policies and procedures, acquisition methods and strategies, project size and complexity, system requirements and development methods, among other things, influence how a system is acquired, developed, operated, or maintained.

To help establish these policies and procedures, and to determine the resources needed by the organization, International Standards can be used to provide specific standardized processes for use within one or more life cycle models. However, in the interest of cost reduction and quality improvement, the processes from the International Standards, as well as life cycle models, may be adapted for an individual project to reflect the variations appropriate to the organization, project and system. The term *tailoring* has grown into use for this adaptation step, so it will be used henceforth in this Technical Report. The two International Standards provide five primary mechanisms for adaptation:

Process selection: Claims of conformance to either standard are made for a declared set of processes. Neither an organization nor a particular project is required to use every process. They may select the processes relevant to their needs and declare that subset as the basis of conformance.

Process substitution: Many of the processes of ISO/IEC 12207 are described as "specializations" of processes in ISO/IEC 15288. Explicit permission is granted to use the system-level process as a basis for conformance rather than the software-level process.

Use of Outcomes: According to the conformance clause of both standards, "conformance is achieved by demonstrating that all of the requirements of the declared set of processes have been satisfied using the outcomes as evidence." The fact that outcomes are used as evidence of conformance provides that alternative selections of activities and tasks may be performed if the outcomes of the declared set of processes are achieved.

Use of Notes: Both standards use non-normative notes or other forms of guidance for provisions that are not required for conformance. In specific situations, implementation of selected notes will be appropriate.

Process tailoring: In both standards, *tailoring* of processes is permitted. Tailoring is defined as the deletion of selected outcomes, activities, or tasks.

NOTE: Tailoring is to be regarded as a last resort form of adaptation. If tailoring is performed, one may only claim tailored conformance rather than full conformance. Furthermore, the deletion of outcomes, activities or tasks may endanger downstream processes by depriving them of necessary prerequisites.

6.2 Adaptation sequence

Figure 16 gives an illustrative sequence of steps that can be followed to adapt the standards to a particular need. First, the adaptations reflecting the project environment are addressed, and then inputs on possible changes are solicited from potentially affected parties. After that, the specific life cycle model and processes are selected. Finally, if necessary, the processes may be tailored, after which the tailoring decisions and rationale are documented. More specifics and examples for systems and software, respectively, can be found in Technical Reports ISO/IEC TR 19760 and ISO/IEC TR 15271.

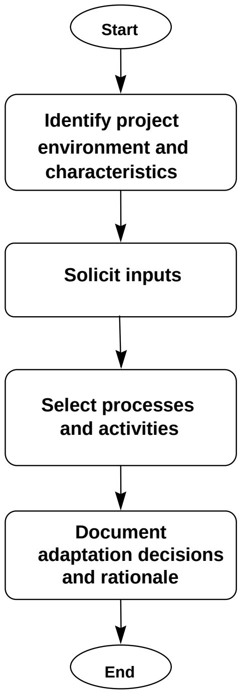

Figure 16—Adaptation sequence

6.2.1 Identify the project environment and characteristics

Organizational characteristics may be determined by considering such issues as organizational policies and procedures. Identify the relevant policies and procedures of the organizations involved, particularly of the acquirer and supplier, with which the project needs to comply. Examples include policies and procedures related to: security, safety, privacy, risk management, use of an independent verification and validation agent, use of a specific computer programming language and, hardware resourcing. Identify any pertinent laws and regulations including those related to environment, public safety and privacy that may impact the project. It may be necessary to monitor such external sources on an ongoing basis to ensure that the system is compliant.

6.2.2 Solicit inputs

The requirements derived from the relevant business and contractual needs are a major driver in adapting the life cycle model and processes from an International Standard. Affected parties should be involved in the adaptation decisions. These people can help ensure that the resulting adapted processes are feasible and useful. Where possible, include feedback from previous projects. For example, ISO/IEC 12207 could be adapted according to the contract between a supplier and the purchaser of a software product. A customer may only require the design of software to be carried out and not full development of a software system. In another instance, if the customer requirements are for safety critical software instead of consumer software, it might be appropriate for the acquirer to require the execution of activities and tasks beyond those provided in ISO/IEC 12207.

6.2.3 Select the appropriate standards

ISO/IEC 15288 and ISO/IEC 12207 provide an interoperable suite of processes suitable for the life cycles of both system and software products. Projects may vary in their relative emphasis upon system and software aspects. This variation may inform the choice of standards appropriate to the project:

- For system projects that have only minor software content, the use of the processes of ISO/IEC 15288 may be sufficient. In this case, ISO/IEC 12207 need not be applied.

- For software projects that have only minor system-level content (perhaps only the process which the software executes), the use of the processes of ISO/IEC 12207 may be sufficient. In this case, ISO/IEC 15288 need not be applied. Furthermore, some of the Technical Processes (ISO/IEC 12207:2008, clause 6.4) may be unnecessary.

- For projects that have both systems and software content, the use of both standards may be merited. It should be noted that ISO/IEC 12207 contains many processes that are specializations of corresponding processes in ISO/IEC 15288. In such cases, one may select the process from ISO/IEC 15288 in preference to the process in ISO/IEC 12207. One criterion for selecting the source of the process may depend on the acquisition strategy. For example, if the system-level prime contractor is performing all configuration management, the prime might require all software subcontractors to conform to the configuration management process of ISO/IEC 15288 rather than the software configuration management process of ISO/IEC 12207. When system-level processes are substituted for software processes, it is appropriate to clarify the role of the software implementers in supporting the execution of the system-level process.

6.2.4 Select life cycle model

Determine which life cycle model(s) are relevant and applicable for the project, such as Waterfall, evolutionary, builds, pre-planned product improvement, Spiral. All such models prescribe certain processes and activities that may be performed sequentially, repeated, and combined; in these models, the life cycle activities in this International Standard should be mapped to the selected model(s). For evolutionary, build, and pre-planned product improvement models, the outputs of one project activity feed into the next. In these cases, the documentation should be complete at the end of an activity or a task. (Annex A provides guidance on development strategies and build planning.)

6.2.5 Select stages and processes

Identify the relevant life cycle model stages that need to be distinguished for the project, as well as their entry and exit criteria and their relationship (serial, parallel, wholly or partly combined). Define the outcomes for each of the stages and the milestones or decision gates that distinguish them. Also, select and prioritize the processes of the appropriate International Standard(s) that will be implemented to achieve the outcomes of the stages.

NOTE: In implementing a suite of processes intended to be applied uniformly throughout an organization, it is usually preferable to start with those processes that will achieve the most significant returns, rather than attempting to implement all of the International Standard at once.

International Standards such as ISO/IEC 12207 or ISO/IEC 15288 do not define the sequencing of processes and activities and they do not prescribe any particular life cycle model. Mapping the organization's current processes, practices and/or methods to the processes, activities and tasks of the applicable International Standard is useful at this stage. The mapping may be used to verify the completeness of the approach; that is, to identify where gaps exist between the current situation and the target situation where processes from the International Standard are used.

6.2.6 Document the adaptation decisions and rationale

When applying International Standards such as ISO/IEC 12207 or ISO/IEC 15288, a mapping of the defined processes and activities onto the selected life cycle model should be documented, together with the determined relationships and the reasons for adopting this approach. The verification of this work is to demonstrate that the outcomes of each stage will be achieved by the processes selected to implement the stage. This documentation should be incorporated into the project management plan for implementing the applicable International Standard as it provides a reference framework for evaluating the success or otherwise of the approach taken.

Specific guidance and examples of adaptation can be found in Technical Reports ISO/IEC TR 15271 and ISO/IEC TR 19760, the application guides for International Standards ISO/IEC 12207 and ISO/IEC 15288, respectively.

6.3 Adaptation guidance

This clause provides general guidance on adapting applications of International Standards such as ISO/IEC 12207 and ISO/IEC 15288 and is not exhaustive. This clause may be used to perform first-level adaptation of the International Standards for a given business area or organization; for example, aviation, nuclear, medical, military, government agency. The second-level adaptation should be performed for each specific project or contract. In general, it is wise for organizations to select and implement processes without tailoring; tailoring should be reserved for the application of the processes to specific projects, and then only as a last resort when other means of adaptation are insufficient to achieve a cost-effective result. Refer to Technical Reports ISO/IEC TR 15271 and ISO/IEC TR 19760 for more specific guidance.

When an International Standard such as ISO/IEC 15288 or ISO/IEC 12207 is used by an organization to form a set of policies and procedures governing project work, then adaptation may be used to appropriately reduce or extend the scope of the International Standard as necessary for the business strategy and kind of business for which the policies and procedures are framed.

When the International Standard is used by a project, then adaptation may be used to appropriately consider the particular characteristics of the project, life cycle stage or agreement. Since each project has to consider and demonstrate the benefits of what it does to satisfy stakeholder requirements, there is a need to concentrate on the relevant processes and activities and the expected outcomes, including specific output documentation.

The International Standard can be considered for a specific project with a set duration or for a continuous work effort conducted by an organization.

When adaptation is done, it may be important to ensure that applicable conformance requirements of the International Standard are met.

Either the organizational unit responsible for forming policies and procedures, or the project team or individual assigned to plan the project can be responsible for completing appropriate adaptation. In any event, all parties involved in the project should be involved in adaptation.

To aid adaptation, the following factors affecting the project effort should be considered:

a) Determine which organizational policies are relevant and applicable, such as on computer languages, safety and security, hardware reserve requirements, and risk management.

b) Determine which acquisition strategies are relevant and applicable for the project, such as types of contract, more than one contractor, involvement of subcontractors and verification and validation agents, level of acquirer's involvement with contractors, and evaluation of contractors' capabilities.

c) Determine which support concepts are relevant and applicable, such as expected length of support, degree of change, and whether the acquirer or the supplier will support. If the software product will have a long support life or is expected to change significantly, all documentation requirements should be considered.

d) Project requirements such as the required work, schedule, funding and technical requirements (for example functional requirements, performance requirements and interface requirements) can drive stage timing and the definition of the system under consideration. These can also drive the criticality of the system and its enabling systems.

e) Project size and criticality factors can drive the amount of controls appropriate for the processes used. A large project involving tens or hundreds of people represents a significant management problem compared with a project of, say, three people. Large projects or projects with subcontractors require careful management supervision and control of a diverse set of parties with varying characteristics. Some projects achieve this through joint reviews, audits, verification, validation and quality assurance. For small projects, all these controls may be excessive. Similarly, the more dependent the system is upon the product operating correctly and being finished on time, the more visibility and control are needed. Conversely, extreme overseeing and control of non-critical parts of the product is probably not cost-effective.

f) The applicable processes of the International Standard that apply to the domain, business and type of organization (for example supplier, user, acquirer, or other stakeholder) should be included in project plans. Other processes that are not in the International Standard can be required by an agreement, or they can be required by the nature of the project, the applicable system or the type of organization. These processes may be added, complete with their purpose, outcomes and activities.

g) The extent of adaptation and the use of the appropriate International Standard as a requirement or guide depend upon the relative position of the project within the life cycle model. Identifying the current point in the life cycle model helps to determine, for example, whether the appropriate International Standard would be used to select methods for conducting computer modelling and simulation in an early stage, or how the International Standard might be adapted in a later stage to address operation, or maintenance of the product.

h) Activities for each applicable process and the expected outcomes of each activity should be selected. Depending on the size and scope of the project, the type of organization and whether an unprecedented system is the object of the project, one or more of the International Standard activities for a process could possibly not apply. Likewise, outcomes and activities may be added to a process when needed to meet agreement requirements or to meet unique requirements for a system.

i) Tasks, methods and tools required for activity completion should be determined. The applicable tasks, methods and available tools are not included in the International Standard. These may be added by the project or organization during planning for an adopted process. See Technical Reports ISO/IEC TR 15271 and ISO/IEC TR 19760 for sources of additional task detail.

j) Reporting and technical review requirements applicable to the life cycle stage or stipulated in the governing agreement or in organizational policies and procedures should be considered.

k) Project measurement requirement provisions should be included for the collection and reporting of key measures by which project progress will be evaluated.

l) Requirements related to activities and tasks involving specialty engineering and functional disciplines may be integrated in appropriate processes. These processes include requirements (special requirements or critical project and system requirements) and life cycle stage entry or exit criteria (for example safety, security, human factor engineering, design, software development, production, test and logistics). Specialty and functional plans that are needed to ensure completion of project work may be included in work definition.

m) Applicable standards, policies and procedures, regulations and laws can be the source of additional process and activity requirements to add to the work definition, even though not included in the International

Standard process requirements. Some reference standards for special factors that should be designed into architectural solutions are provided in ISO/IEC TR 15271 and ISO/IEC TR 19760.

Additional adaptation considerations can be found in the conformance requirements of the International Standard.

6.4 Scope adaptation

As an example, if an organization does development only and is not involved in the utilization, support, or retirement life cycle stages, that organization could adapt the scope of the applicable International Standard accordingly by selecting only the appropriate processes. The policies and procedures called for in the non-applicable parts of the International Standard would not be included in the organization's policies and procedures. Additionally, inputs such as those listed below can help shape the policies and procedures of an organization:

a) Life cycle model and related entry or exit criteria used by the organization for decision making as well as for establishing milestone reviews of a project. Select one or more appropriate life cycle models for the project, since hardware, software, humans and other aspects of the system may have their own life cycles. Determine whether the software life cycle model is a sub-part of the life cycle model of the system or is the complete life cycle model.

b) Resource availability and the resources the organization is willing to commit;

c) Expertise and skills available to the organization to provide the organization's products and services;

d) Technology available for the organization's products and services.

6.5 Stage adaptation

Depending on the specific system-of-interest and the environment in which a project is established to realize it, stages may be combined, eliminated, or added to the generic model. For example, an organization may buy a concept, establish a project to design and produce a system, then turn the result over to another organization for marketing to consumers, or operation and maintenance. In another case, there could be repeated iterations between concept and development stages, or lengthy evolution after fielding of a system. Each stage in these cases would have different criteria for entry and exit, as well as possibly drawing on different processes. Accordingly, each stage in the life cycle, as well as the juxtaposition of each stage with those before and after it, requires specific consideration for the project and system to be realized.

6.6 Process adaptation

When a project is established to satisfy a set of stakeholder requirements or acquirer specifications, processes included in the organizational policies and procedures, or in the applicable International Standard itself, can be adapted according to the scope, size and funding of the work to be done, as well as: the team structure; requirements and schedules; the specific life cycle model used and; project resources. In some cases, outcomes, activities or tasks, suggested by notes or other sources, may be added. In some cases, outcomes, activities or tasks may be deleted via tailoring.

The technical processes need special attention, because the process may be used by different parties with different objectives. For example, if a product is embedded in or integral to the system, all the activities in the processes should be considered and it should be specifically agreed who is required to *perform or support* each processes activities.

Also, development of software product may have special technical risks. If the software technology used is not mature, software product being developed is unprecedented or complex. If a software product contains safety, security, or other critical requirements, then rigorous specification, design, testing, and evaluations may be needed. In those cases, verification and validation should be independent and address the complete product.

6.7 Adapting evaluation-related activities

Persons who are involved in any activity of the life cycle of a project or a process may conduct evaluations either on their own or other's products and activities. This Technical Report groups these evaluations into five categories, which are listed below. The first four evaluation categories are at the project level; the last one is at the organizational level. These evaluations should be selected and adapted proportional to the scope, magnitude, complexity, and criticality of the project or of the organization. The problem, non-conformance, and improvement reports from these evaluations feed back into one or more of the life cycle processes shown in Figure 10

a) Process-internal evaluations are conducted by personnel performing the assigned tasks within the process during their day-to-day activities.

b) Verification and Validation 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.

c) Joint reviews and audits are conducted in a joint forum by the reviewing and reviewed parties to evaluate status and compliance of products and activities on a pre-agreed to schedule. Additional guidance on joint reviews is given in Annex B.

d) Quality management is conducted by personnel independent of the personnel directly responsible for developing the product or executing the process. The goal is to *independently assure* conformance of the products and processes with the contract requirements and adherence to the established plans. This process may use the results from a, b, and c above as inputs. This process may coordinate its activities with those of a, b, and c.

e) Improvement is conducted by an organization for efficient management and self-improvement of its process. This is conducted regardless of project or contract requirements.

7 Life cycle model use by domains, disciplines and specialties

7.1 Life cycle models for domains and disciplines

The life cycle model that is appropriate at the system level may need adaptation if the system-of-interest falls largely or entirely within one domain, such as software. Further, the life cycle model that best reflects events in one domain (such as software) may not be equally suitable for others (such as hardware, humans, processes, procedures, facilities and naturally occurring entities), as is illustrated by the examples shown in Figure 17.

The life cycle stages shown are illustrative only, this is not an exhaustive list of possible life cycle models. For example, using life cycle models is just as relevant to services as it is to hardware, software, or other products. A life cycle model for service management could include stages of: Service Strategy; Service Design; Service Transition Service Operation and; Continual Service Improvement, as shown in Figure 17. The advantage to a service organization of using a life cycle model approach is that it gives them a unifying framework for examining what processes they need to have to do their work. If a service management organization viewed itself as doing nothing but service operations, not thinking in terms of a life cycle model, it could, for example, fail to consider continual service improvement, or not occasionally re-examine its service strategy. That could make the organization less desirable to a customer compared to a service management organization that uses a life cycle model framework to keep its competitive view as wide as possible.

Note that, for all models, the length of any one stage, or of the entire model, does not represent any uniform time scale or coincident start or stop timing.

The basic point is that each domain should be considered as an entity that can have a series of stages through which it goes, forming a life cycle model for that domain. As a result, when a system has elements that span multiple domains:

a) Every domain's life cycle model should be thought through, and;

b) The life cycle models and their stages must be considered as a whole and care taken that they work in concert.

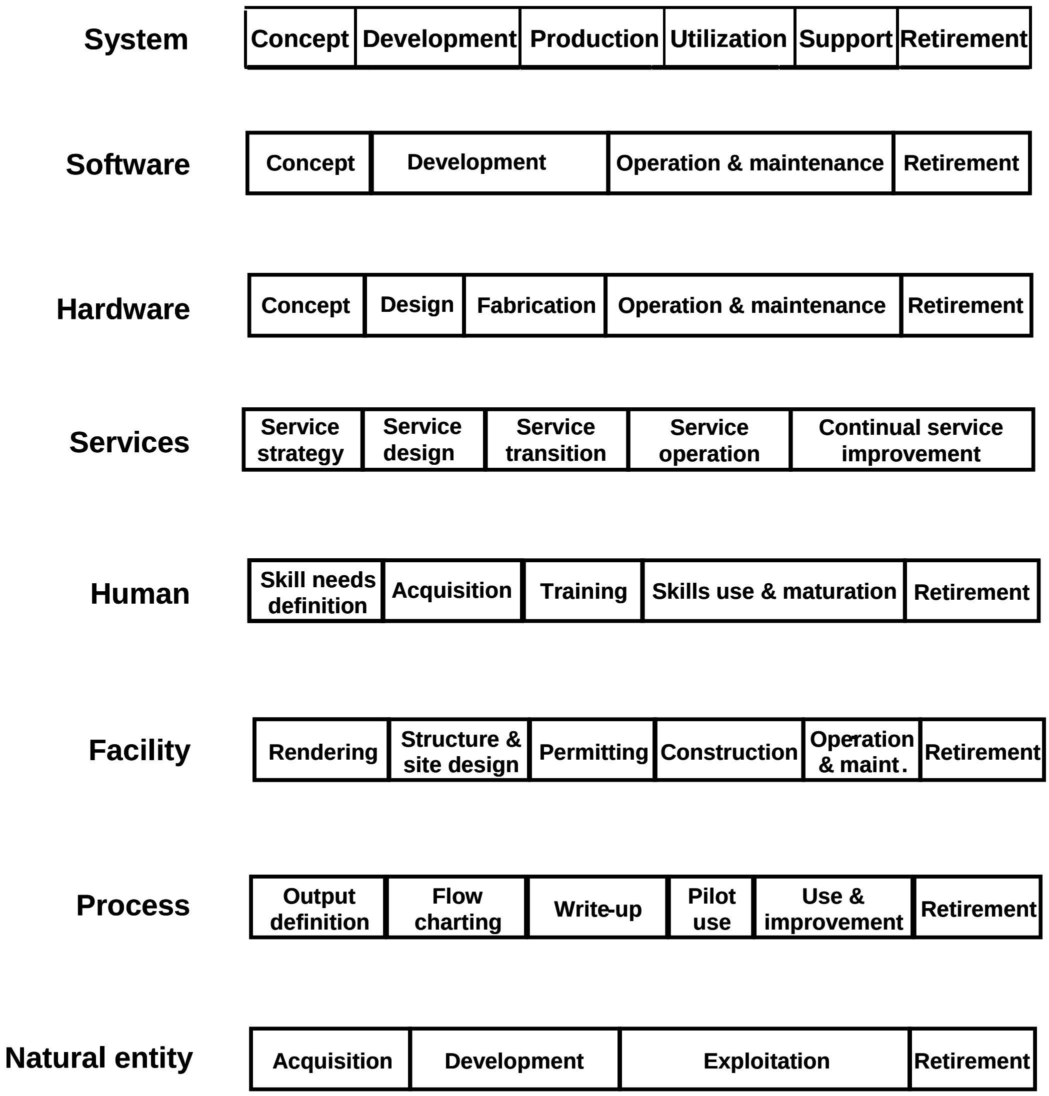

Figure 17—Illustrative examples of domain life cycle models

7.2 Adaptation for domains and disciplines

In general, life cycle models would not need to be developed or adapted for a specific discipline (such as mechanical engineering, electrical engineering, civil engineering, quality, system administration). Instead, the processes associated with the life cycle model(s) in which that discipline is used would be adapted to reflect the overall considerations discussed in clause 6 of this Technical Report, possibly with additional adaptation for the discipline itself. For example, the agreement processes associated with a facility would be adapted to the specific facility project, as described in clause 6. In addition, some details might be adapted to reflect civil engineering concepts, practices, or terminology.

7.3 Adaptation for specialties

The focus of International Standards is on the engineering, operation, maintenance and disposal of complex man-made products. Each product can have critical qualities that should be considered, with possible adaptation of specific processes, so that the product can be successful. The list of these critical qualities is long and not well agreed on and could include, but not be limited, to such areas as: human factors, health and safety, reliability, maintainability, availability, supportability/usability, security, environmental impacts (including disposability), electromagnetic compatibility, mass properties, interoperability. These critical qualities, although requiring special knowledge and expertise in one area, generally can not be evaluated in isolation from each other. So, for example, assessments of safety, human factors, and environmental compatibility may need to be done as one integrated effort and that assessment could link to yet others. Some, though obviously not all, of the critical qualities are discussed below to illustrate adaptation for specialities.

7.3.1 Human

Human interaction with products and services associated with a system should be looked at from the perspective of impacts on operators, users and the general public. Impacts should be analyzed to determine adverse impacts that should be avoided or mitigated through product related design requirements that could mitigate the adverse impacts identified.

7.3.2 Health

Planned usage rates and environments, operational concepts and other requirements can present health risks with respect to potential damage to human life including operators and others (both people and animals) that come in contact with the product or exist within its operational environment. Use cases, human-machine interfaces, operating environments, electromagnetic radiation, heat and noise emissions and waste materials should be analysed to determine such risks. Outcomes from such analyses should include specific health concerns and recommendations as to health related design requirements that could prevent the health hazards identified.

Note that health issues may persist after the product is retired.

7.3.3 Safety

Operational concepts and other product requirements can present safety risks with respect to potential damage to human life, property and the environment. Use cases, human-machine interfaces, operating environments, electromagnetic radiation, heat and noise emissions, waste materials and failure modes should be analysed to determine such risks. Outcomes from such analyses should include specific safety concerns and recommendations as to safety related design requirements that could prevent the safety hazards identified.

7.3.4 Security

Operational concepts, usage environments and other product requirements can present security risks with respect to the product and its users. Risks include 1) access and damage to personnel, properties and information, 2) corruption, theft or compromise of sensitive information, 3) denial of approved access to property and information, 4) unauthorized system access and 5) loss of life or property. Applicable areas of security should be analysed to include physical security, communications security, computer security and electronic emissions security. Outcomes from such analyses should include specific security concerns and recommendations as to security related requirements that could mitigate the security risks identified.

7.3.5 Interoperability

Data flows are essential within a product and possibly between products so that operational functions can be successfully performed over the product's life. The potential failure causes of data (or information) to properly flow should be analysed to include use of appropriate communication connectivity protocols with external systems or internal systems within a system structure. Outcomes from such analyses should include specific interoperability concerns and recommendations as to related design requirements that could improve interoperability.

7.3.6 Usability

The operational effectiveness and hence the acceptance of many products depends on a user's ability to realize the intended capability of the product. Systems that include human elements depend on operators of the product performing tasks within specified times and required accuracies and with efficient and effective resource utilization. Use cases, human-machine interfaces, operating environments and training and operating procedures should be defined based on targets for usability such as understandability, learnability, operability and attractiveness, and evaluated against quality of use criteria such as effectiveness, productivity, safety and satisfaction.

7.3.7 Dependability

The failure potential of any part of the product will determine whether the product will be available when and as long as it is needed during any operational use and at any given (random) time. Factors that affect dependability include mean-time-between-failures, mean-time-to-repair and administrative down time. Such factors should be analysed to determine their impact over the product life. Outcomes from such analyses should include specific dependability concerns and recommendations as to related requirements that could help make the product more dependable.

7.3.8 Environmental impacts

The impacts on the environment from short and long-term use of a product, and disposing of hazardous materials related to its use or retirement, can present risks to all life forms. Risks include loss of life, illness and lowering of the standard of living. Environmental impacts as a result of use, or disposal of waste products from product use or from disposal of the product or one of its elements that have reached end-of-life, should be analysed. Outcomes from such analyses should include specific environmental impact concerns and recommendations as to related requirements that could reduce risks related to product use and eventual disposal.

8 Relationship with detailed process standards

This Technical Report gives overall guidance on life cycles and adaptation that can be applied across the International Standards (the conformance documents) that could be applied to a particular situation. Further guidance on the application of the processes of each applicable International Standards would come from other Technical Reports. These relationships are illustrated in Figure 18.

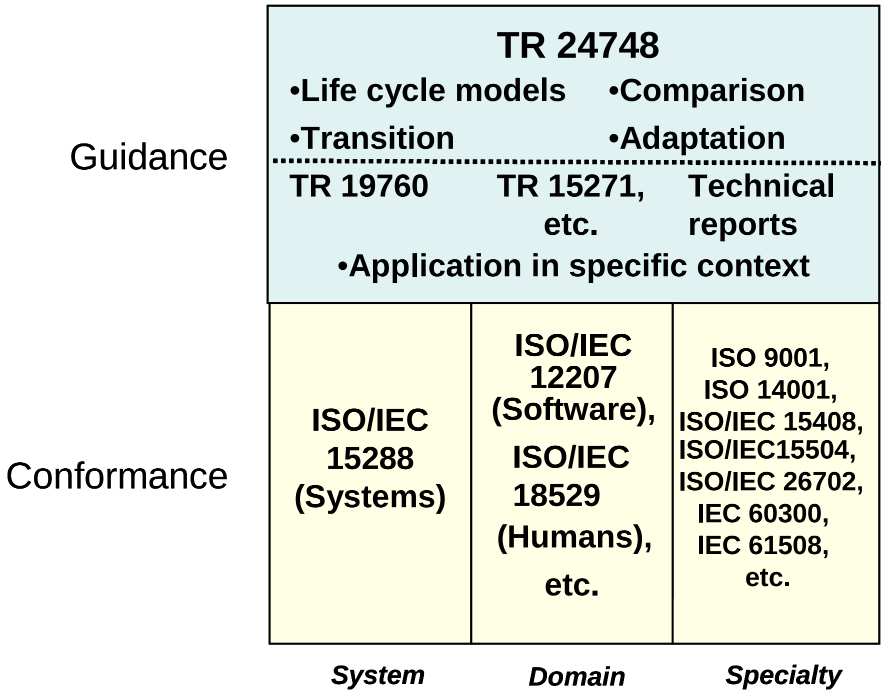

Figure 18—ISO/IEC 24748 relationship to detailed process standards

International Standard ISO/IEC 15288 defines a generic, top-level framework based upon a set of processes that can be combined into various suitable life cycle models. It does not, and is not intended to, define in detail systems engineering or the engineering of systems. However, the International Standard is expected to strengthen the relationships among systems engineering, software engineering and other affected engineering disciplines. It is intended to do this through promotion of consistent and uniform terminology among the various domains and engineering disciplines. It is also intended to establish interactions and improved communication between the various engineering disciplines needed to create systems.

The other International Standards address either domains (e.g. software, humans) in detail, or treat specific disciplines or specialties. The International Standards are used as the basis for building applicable sets of life cycle processes that provide activities to achieve a stated goal. The processes defined by the International Standards are likely to be invoked during the whole life cycle of the system. The International Standards are, in turn, supported by application guidelines, such as ISO/IEC TR 15271 for software. This Technical Report thus provides spanning guidance in two key areas of interest across domains and disciplines, complemented by the specific conformance and guidance documents for each area. A more detailed view of the emphasis given each area (e.g. process definitions) by each of the International Standards and the related Technical Reports is shown in Figure 19. For example, ISO/IEC 15288 gives process definitions for systems engineering processes, including those also system-level processes also used in ISO/IEC 12207. ISO/IEC 12207 repeats those process definitions and adds processes specific to software. ISO/IEC TR 24748 gives a general overview of the process definitions, as background for treatment of life cycle processes, while the other two technical reports take the process sets as givens.

Area	ISO/IEC 15288	ISO/IEC 12207	ISO/IEC TR 24748	ISO/IEC TR 19760	ISO/IEC TR 15271
Process definitions	Systems engineering & common	Software engineering	General overview: what a process is & pointer to standards	n/a	n/a
Life cycle concepts	Summary	Summary	Detail	n/a	n/a
Life cycle stages	Summary	Summary	Detail	n/a	n/a
Life cycle tailoring	Process requirements	Process requirements	General guidance	Specific detail for systems engineering	Specific detail for software engineering
Life cycle application/usage	n/a	n/a	General guidance	Domain-specific guidance	Domain-specific guidance
Life cycle model examples/illustrations	n/a	n/a	Both a system & software example	Domain-specific examples	Domain-specific examples
Terminology	Systems engineering	Software engineering	Life cycle & pointer to standards	As needed	As needed
System process key concepts	Summary	n/a	n/a	Detail	n/a
Software process key concepts	n/a	Summary	n/a	n/a	Detail
Organization/project application	Summary	Summary	Detail	n/a	n/a
Process application	Summary	Summary	General overview	Detail	Detail
Process tailoring	Normative requirements	Normative requirements	General guidance	Example for systems	Example for software
Process reference model	Detail	Detail	General description & pointer to standards	n/a	n/a
Specialty applications	Summary	Summary	General tailoring introduction	Detail for systems	Detail for software
Conformance	Included	Included	n/a	n/a	n/a

Figure 19—Overview of coverage and emphasis among standards and technical reports

9 Guidance on transitioning from the previous versions

International Standards ISO/IEC 12207:1995 and ISO/IEC 15288:2002 have differences in terminology, as well as in process names, structure and content, even for processes that address the same area. To help the communication between the two domains, as well as joint usage of the two documents in a systems and software environment, the revisions to ISO/IEC 15288:2008 and ISO/IEC 12207:2008 align terminology, process names and process structure. Revision is planned to further rationalize the process detail and overall content of the two International Standards. Because of these changes, current and potential users of either or both document(s) face a decision on when to adopt which version of each standard. While those decisions can only be made by each user in the specific context of their technical and business environment, a more informed decision can be made if the user has more information on the current and future situation regarding the International Standards. It is the purpose of this clause to provide that information.

9.1 Comparisons between the versions

The changes between the original versions of ISO/IEC 12207 (i.e. 12207:1995) and ISO/IEC 15288 (i.e. 15288:2002) and the 2008 updates are shown in this clause. There are several ways in which the two standards have changed. The first is the overall process structure, shown in Figure 20. ISO/IEC 12207:1995 and its amendments did not have the same process structure.

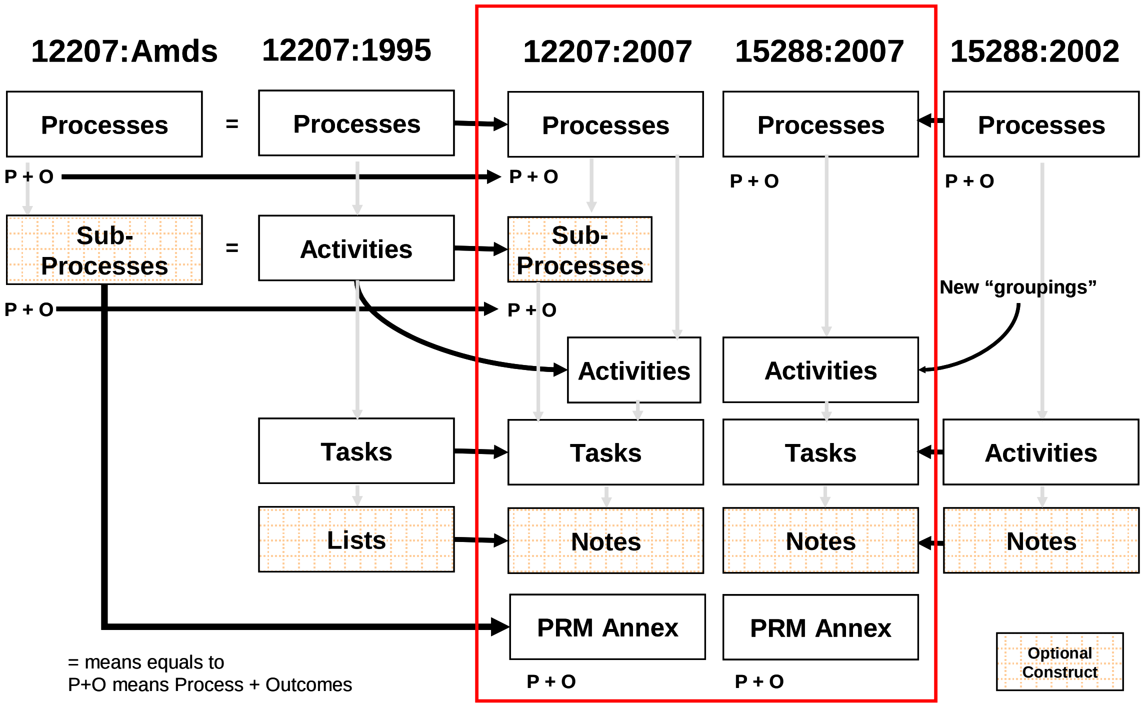

Figure 20—Process structure changes of the 2008 updates

Additionally, the structure of the ISO/IEC 12207:1995 processes was different from that of ISO/IEC 15288:2002. In the 2008 revisions, both International Standards have a common structure of Process Purpose, Process Outcomes, Activities and Tasks. ISO/IEC 12207:2008 also contains lower-level processes to further elaborate complex processes. In addition, both International Standards include a Process Reference Model to support assessment needs. The process purposes and outcomes for processes with the same names have been made the same wherever possible. This does not mean that the process contents are the same between the two International Standards. There are also processes where the names, while not identical, are more similar than before.

For the benefit of the users of the previous edition of ISO/IEC 12207 and its Amendments and the previous edition of ISO/IEC 15288 there are several mappings from the original set of normative process clauses to the revised international standards provided in Figures 21 through 25.

Figure 21 shows the mapping between the ISO/IEC 15288:2002 and ISO/IEC 15288:2008 sets of normative process clauses.

ISO/IEC 15288:2002		ISO/IEC 15288:2008		
Clause	**Process**	**Clause**	**Process**	**Changes**
5	**System life cycle processes**	**6**	**System life cycle processes**	
5.1	Introduction			Deleted
5.2	Agreement processes	6.1	Agreement processes	Numbering
5.2.1	Introduction	6.1	Agreement processes	Combined into 6.1; deleted separate clause
5.2.2	Acquisition process	6.1.1	Acquisition process	Numbering
5.2.3	Supply process	6.1.2	Supply process	Numbering
5.3	Enterprise processes	6.2	Organizational project-enabling processes	Topic and title revised; numbering
5.3.1	Introduction	6.2	Organizational project-enabling processes	Combined into 6.2; deleted separate clause
5.3.2	Enterprise environment management process	6.2.2	Infrastructure management process	Topic and title revised; numbering
5.3.3	Investment management process	6.2.3	Project portfolio management process	Topic and title revised; numbering
5.3.4	System life cycle processes management process	6.2.1	Life cycle model management process	Title revised; numbering
5.3.5	Resource management process	6.2.2	Infrastructure management process	Separated human resources from other resources; numbering
		6.2.4	Human resource management process	
5.3.6	Quality management process	6.2.5	Quality management process	Numbering
5.4	Project processes	6.3	Project processes	Numbering
5.4.1	Introduction	6.3	Project processes	Combined into 6.3; deleted separate clause
5.4.2	Project planning process	6.3.1	Project planning process	Numbering
5.4.3	Project assessment process	6.3.2	Project assessment and control process	Processes merged; numbering
5.4.4	Project control process			
5.4.5	Decision-making process	6.3.3	Decision management process	Title revised; numbering
5.4.6	Risk management process	6.3.4	Risk management process	Numbering
5.4.7	Configuration management process	6.3.5	Configuration management process	Numbering
5.4.8	Information management process	6.3.6	Information management process	Numbering
		6.3.7	Measurement process	Added
5.5	Technical processes	6.4	Technical processes	Numbering
5.5.1	Introduction	6.4	Technical processes	Combined into 6.4; deleted separate clause

5.5.2	Stakeholder requirements definition process	6.4.1	Stakeholder requirements definition process	Numbering
5.5.3	Requirements analysis process	6.4.2	Requirements analysis process	Numbering
5.5.4	Architectural design process	6.4.3	Architectural design process	Numbering
5.5.5	Implementation process	6.4.4	Implementation process	Numbering
5.5.6	Integration process	6.4.5	Integration process	Numbering
5.5.7	Verification process	6.4.6	Verification process	Numbering
5.5.8	Transition process	6.4.7	Transition process	Numbering
5.5.9	Validation process	6.4.8	Validation process	Numbering
5.5.10	Operation process	6.4.9	Operation process	Numbering
5.5.11	Maintenance process	6.4.10	Maintenance process	Numbering
5.5.12	Disposal process	6.4.11	Disposal process	Numbering
Annex A	Tailoring process	Annex A	Tailoring	Title revised

Figure 21—Mapping of process clause sets between ISO/IEC 15288:2002 and ISO/IEC 15288:2008

Figures 22 through 25 include treatments for ISO/IEC 12207 processes and sets of process clauses. The information in these tables should be used with caution because:

— The entries in the tables could only be made approximate rather than precise. Each standard needs to be reviewed in detail for adequate understanding of any changes.

— Provisions were sometimes adapted, occasionally extensively, for better fit into their new context.

— The text of the provisions may have changed during the process of reaching consensus.

— Figure 22 provides information regarding the source of the provisions in the aligned process clause set of ISO/IEC 12207:2008. See below the table for abbreviations used.

Clause	Process	Source of purpose & outcomes	Source of activities & tasks
6.1	**Agreement processes**		
6.1.1	Acquisition	Amd 1, F.1.1	12207:1995, 5.1; 15288, 5.2.2.3
6.1.2	Supply	Amd 1/Amd 2, F.1.2	15288 5.2.3.3 (a, h, i); 12207:1995, 5.2
6.2	**Organizational project-enabling processes**		
6.2.1	Life cycle model management	15288, 5.3; Amd 1, F.3.3	12207:1995, 7.3
6.2.2	Infrastructure management	15288, 5.3; Amd 1, F.3.2	12207:1995, 7.2
6.2.3	Project portfolio management	15288, 5.3; Amd 1, F.3.1.1	15288, 5.3.3
6.2.4	Human resource management	15288, 5.3.5; Amd 1, F.3.4	Amended 12207, 7.4
6.2.5	Quality management	15288, 5.3.6; Amd 1, F.3.1.4	15288, 5.3.6

Clause	Process	Source of purpose & outcomes	Source of activities & tasks
6.3	**Project processes**		
6.3.1	Project planning	15288, 5.4.2; Amd 1, F.3.1.3	12207:1995, 7.1.1, 7.1.2 &7.1.3.1
6.3.2	Project assessment and control	15288, 5.4.3 and 5.4.4; Amd 1, F.3.1.3 (4), (6), (7)	12207:1995, 7.1.3.2 through 7.1.3.4; 7.1.4; 7.1.5
6.3.3	Decision management	15288, 5.4.5	15288, 5.4.5
6.3.4	Risk management	16085, 5; Amd 1, F.1.3.5	16085, 5
6.3.5	Configuration management	15288, 5.4.7	15288, 5.4.7
6.3.6	Information management	15288, 5.4.8	15288, 5.4.8
6.3.7	Measurement	15939, 4.1; Amd 1, F.1.3.6	15939, 4 and 5
6.4	**Technical processes**		
6.4.1	Stakeholder requirements definition	Amd 1, F.1.3.1	15288, 5.5.2
6.4.2	System requirements analysis	Amd 1, F.1.3.2	12207:1995, 5.3.2
6.4.3	System architectural design	Amd 1, F.1.3.3	12207:1995, 5.3.3
6.4.4	Implementation	Not applicable	Not applicable
6.4.5	System integration	Amd 1, F.1.3.9	12207:1995, 5.3.10
6.4.6	System qualification testing	Amd 1, F.1.3.10	Amended 12207, 5.3.11
6.4.7	Software installation	Amd 1, F.1.3.11	12207:1995, 5.3.12
6.4.8	Software acceptance support	Amd 2, F.1.2.4	12207:1995, 5.3.13
6.4.9	Software operation	15288, 5.5.10; Amds 1 & 2, F.1.4	12207:1995, 5.4
6.4.10	Software maintenance	Amd 1, F.1.5	12207:1995, 5.5.1 through 5.5.5
6.4.11	Software disposal	15288 5.5.12; Amd 1, F.1.5 (6)	12207:1995, 5.5.6
7.1	**Software implementation processes**		
7.1.1	Software implementation	15288, 5.5.5.1	Amended 12207, 5.3.1
7.1.2	Software requirements analysis	Amd 1, F.1.3.4	Amended 12207, 5.3.4
7.1.3	Software architectural design	Amd 1, F.1.3.5	12207:1995, 5.3.5
7.1.4	Software detailed design	Amd 1, F.1.3.5	12207:1995, 5.3.6

Clause	Process	Source of purpose & outcomes	Source of activities & tasks
7.1.5	Software construction	Amd 1, F. 1.3.6	12207:1995, 5.3.7
7.1.6	Software integration	Amd 1, F.1.3.7	12207:1995, 5.3.8
7.1.7	Software qualification testing	Amd 1, F.1.3.8	12207:1995, 5.3.9
7.2	**Software support processes**		
7.2.1	Software documentation management	Amd 1, F.2.1	12207:1995, 6.1
7.2.2	Software configuration management	Amd 1, F.2.2	12207:1995, 6.2
7.2.3	Software quality assurance	Amd 1, F.2.3	Amended 12207, 6.3
7.2.4	Software verification	Amd 1, F.2.4	12207:1995, 6.4.
7.2.5	Software validation	Amd 1, F.2.5	12207:1995, 6.5
7.2.6	Software review	Amd 1, F.2.6	12207:1995, 6.6
7.2.7	Software audit	Amd 1, F.2.7	12207:1995, 6.7
7.2.8	Software problem resolution	Amd 2, F.2.8	12207:1995, 6.8
7.3	**Software reuse processes**		
7.3.1	Domain engineering	Amd 1, F.3.7	Amd 1, G.6
7.3.2	Reuse asset management	Amd 1, F.3.5	Amd 1, G.4
7.3.3	Reuse program management	Amd 1, F.3.6	Amd 1, G.5
Annex A	Tailoring process	Added new. Parallels 15288:2008, Annex A, A.2.1, A.2.2	Added new. Parallels 15288:2008, Annex A, A.2.3
Annex B	Process Reference Model (PRM) for Assessment Purposes	Added new using 12207:2008 clauses 6 and 7	Not applicable

Figure 22—ISO/IEC 12207:2008 Process Definition Sources

The sources cited in the table are the following:

- "12207:1995" refers to ISO/IEC 12207:1995.
  - "Amd 1" refers to the annexes added by ISO/IEC 12207:1995/AMD.1:2002.
  - "Amd 2" refers to the annexes added by ISO/IEC 12207:1995/AMD.2:2004.
  - "Amds 1 & 2" refers to the annexes added by AMD.1 as amended by AMD.2.

- "Amended 12207" refers to the main body text of ISO/IEC 12207:1995 as amended by AMD.1 and AMD.2.
- "15288" refers to ISO/IEC 15288:2002.
- "15939" refers to ISO/IEC 15939:2002.
- "16085" refers to ISO/IEC 16085:2004.

Figure 23 provides the corresponding mapping between ISO/IEC 12207:1995 and ISO/IEC 12207:2008 process clause sets and Figure 24 gives the inverse mapping between ISO/IEC 12207:2008 and ISO/IEC 12207:1995.

ISO/IEC 12207:1995*		**ISO/IEC 12207:2008**		**Changes**
Clause	Process/Activity	Clause	Process	
5	Primary life *cycle* processes	6	System Life Cycle Processes	System and software separated; Topic and title revised; numbering
		7	Software Life Cycle Processes	
		6.1	Agreement Processes	Added – new structure
5.1	Acquisition process	6.1.1	Acquisition *Process*	Numbering Acquisition process lower level processes purposes and outcomes in Annex B
5.2	Supply process	6.1.2	Supply Process	Numbering Supply process lower level processes purposes and outcomes in Annex B
5.3	Development process	6.4	Technical Processes	System context activities and software implementations activities separated; Topic and title revised; numbering
		7.1	Software Implementation Processes	

ISO/IEC 12207:1995*		ISO/IEC 12207:2008		Changes
5.3.1	Process implementation	7.1.1	Software Implementation Process	Re-positioned in Software Life Cycle Processes; numbering; renamed
		6.4.1	Stakeholder Requirements Definition Process	Added – adapted from 15288
5.3.2	System requirements analysis	6.4.2	System Requirements Analysis Process	Numbering
5.3.3	System architectural design	6.4.3	System Architectural Design Process	Numbering
5.3.4	Software *requirements* analysis	7.1.2	Software Requirements Analysis Process	Numbering
5.3.5	Software architectural design	7.1.3	Software Architectural Design Process	Numbering
5.3.6	Software detailed design	7.1.4	Software Detailed Design Process	Numbering
5.3.7	Software coding and testing (* renamed Software Construction process by Amd 1)	7.1.5	Software Construction Process	Numbering
5.3.8	Software integration	7.1.6	Software Integration Process	Numbering
5.3.9	Software qualification testing	7.1.7	Software Qualification Testing Process	Numbering
		6.4.4	Implementation Process	Added
5.3.10	System integration	6.4.5	System Integration Process	Numbering
5.3.11	System qualification testing	6.4.6	System Qualification Testing Process	Numbering
5.3.12	Software installation	6.4.7	Software Installation Process	Numbering

ISO/IEC 12207:1995*		ISO/IEC 12207:2008		Changes
5.3.13	Software acceptance support	6.4.8	Software Acceptance Support Process	Numbering
5.4	Operation process	6.4.9	Software Operation Process	Revised (Software Operation context includes operation of system level for the software with system); numbering; renamed Operation process lower level processes purposes and outcomes in Annex B
5.5	Maintenance process	6.4.10	Software Maintenance Process	Numbering; renamed
		6.4.11	Software Disposal *Process*	Clause 5.5 in ISO/IEC 12207:1995 is divided into 6.4.10 and this new clause 6.4.11; renamed
6	Supporting life cycle processes	7.2	Software Support Processes	System and software are separate (System level support processes are included under 6.2 Organizational Project Enabling Processes and 6.3 Project Processes); Topic and title revised;
6.1	Documentation process	7.2.1	Software Documentation Management Process	Numbering; renamed
6.2	Configuration management process	7.2.2	Software Configuration Management Process	System and software are separate; 7.2.2 numbering; renamed 6.3.5 added (adapted from 15288)
		6.3.5	Configuration Management Process	

ISO/IEC 12207:1995*		ISO/IEC 12207:2008		Changes
6.3	Quality assurance process	7.2.3	Software Quality Management Process	System and software are separate; 7.2.3 numbering; renamed; 6.2.5 added (adapted from 15288)
		6.2.5	Quality Management Process	
6.4	Verification process	7.2.4	Software Verification Process	Numbering; renamed
6.5	Validation process	7.2.5	Software Validation Process	Numbering; renamed
6.6	Joint review process	7.2.6	Software Review Process	Numbering; renamed
6.7	Audit process	7.2.7	Software Audit Process	Numbering; renamed
6.8	Problem resolution Process	7.2.8	Software Problem Resolution Process	Numbering; renamed
6.9*	Usability process	E.4 (Annex)	Process view for usability	Usability is not defined as a distinct process in ISO/IEC 12207:2008; Activities for usability are not defined in the independent process, but summarised as considerable issue of each process in Annex E.4 of ISO/IEC 12207:2008.
7 Organizational life cycle processes		6.2	Organizational Project-Enabling Processes	Topic and title revised; Organization and project separated;
		6.3	Project Processes	
		7.3	Software Reuse Processes	
7.1	Management process	6.3.1	Project Planning	Topic and title revised; numbering;

ISO/IEC 12207:1995*		ISO/IEC 12207:2008		Changes
				Management process lower level processes purposes and outcomes in Annex B and in Annex F
		6.3.2	Project Assessment and Control	
		6.3.4	Risk Management Process	added (Detailed at project level) added (adapted from 16085)
		6.3.6	Information Management Process	added (adapted from 15288)
		6.3.7	Measurement Process	added (adapted from 15939)
		6.2.3	Project Portfolio Management Process	Added (adapted from 15288)
		6.3.3	Decision Management Process	added (adapted from 15288)
7.2	Infrastructure process	6.2.2	Infrastructure Management Process	Numbering; renamed
7.3	Improvement *process*	6.2.1	Life Cycle Model Management Process	Numbering; renamed; Improvement process lower level processes purposes and outcomes in Annex B
7.4	Training process (* Amd 1 renamed to Human resource process and added activities and tasks)	6.2.4	Human Resource Management Process	Topic and title revised; numbering; renamed; Human resource process lower level processes purposes and outcomes in Annex B
7.5*	Asset management process	7.3.2	Reuse Asset Management Process	Numbering; renamed

ISO/IEC 12207:1995*		ISO/IEC 12207:2008		Changes
7.6*	Reuse program management process	7.3.3	Reuse Program Management Process	Numbering
7.7*	Domain engineering process	7.3.1	Domain Engineering Process	Numbering

* Including Amendments 1 and 2

Figure 23—Mapping of process clause sets between ISO/IEC 12207:1995 and ISO/IEC 12207:2008

ISO/IEC 12207:2008		ISO/IEC 12207:1995		
Clause	**Process**	**Clause**	**Process**	**Changes**
6	**System Life Cycle Processes**	**5**	**Primary life cycle processes**	Topic, title and structural revisions due to reallocation of Amended 12207 clauses across 12207:2008; See next level clauses
		6	Supporting life cycle processes	
		7	Organizational life cycle processes	
6.1	Agreement Processes			Added structure
6.1.1	Acquisition Process	5.1	Acquisition process	Numbering
6.1.2	Supply Process	5.2	Supply process	Numbering
6.2	Organizational Project-Enabling Processes			Restructured clauses 6 and 7
6.2.1	Life Cycle Model Management Process	7.3	Improvement process	Title revised; Numbering
6.2.2	Infrastructure Management Process	7.2	Infrastructure process	Title revised; numbering
6.2.3	Project Portfolio Management Process			Added (adapted from 15288)

ISO/IEC 12207:2008		ISO/IEC 12207:1995		
Clause	**Process**	**Clause**	**Process**	**Changes**
6.2.4	Human Resource Management Process	7.4	Training process (Amd 1 renamed to Human resource process and added activities and tasks)	Topic and title revised; numbering
6.2.5	Quality Management Process			Added system level (Adapted from 15288)
6.3	Project Processes	7	Organizational life cycle processes	Topic, title and structural revision
6.3.1	Project Planning Process	7.1	Management process	Topic, title and structure revised; numbering
6.3.2	Project Assessment and Control Process			
6.3.3	Decision Management Process			Added (adapted from 15288)
6.3.4	Risk Management Process			Added (adapted from 16085)
6.3.5	Configuration Management Process			Added system level (Adapted from 15288)
6.3.6	Information Management Process			Added system level (adapted from 15288)
6.3.7	Measurement Process			Added (adapted from 15939)
6.4	Technical Processes	5.3 5.4 5.5	Development process Operation process Maintenance process	Topic, title and structure revised; numbering
6.4.1	Stakeholder Requirements Definition Process			Added (adapted from 15288)
6.4.2	System Requirements Analysis Process	5.3.2	System requirements analysis	Numbering
6.4.3	System Architectural Design Process	5.3.3	System architectural design	Numbering

ISO/IEC 12207:2008		ISO/IEC 12207:1995		
Clause	**Process**	**Clause**	**Process**	**Changes**
6.4.4	Implementation Process			Added system level structural concept; Points to Clause 7.1 Software Implementation Processes
6.4.5	System Integration Process	5.3.10	System integration	Numbering
6.4.6	System Qualification Testing Process	5.3.11	System qualification testing	Numbering
6.4.7	Software Installation Process	5.3.12	Software installation	Numbering
6.4.8	Software Acceptance Support Process	5.3.13	Software acceptance support	Numbering
6.4.9	Software Operation Process	5.4	Operation process	Numbering
6.4.10	Software Maintenance Process	5.5	Maintenance process	Restructure, Numbering
6.4.11	Software Disposal Process			
7	Software Life Cycle Processes	5	Primary life cycle processes	Topic, title and structural revisions due to reallocation of Amended 12207 clauses across 12207:2008; See next level clauses
		6	Supporting life cycle processes	
		7	Organizational life cycle processes	
7.1	Software Implementation Processes	5.3	Development process	Restructure, Title Numbering
7.1.1	Software Implementation Process	5.3.1	Process implementation	Restructure, Numbering
7.1.2	Software Requirements Analysis Process	5.3.4	Software requirements analysis	Numbering
7.1.3	Software Architectural Design Process	5.3.5	Software architectural design	Numbering

ISO/IEC 12207:2008		ISO/IEC 12207:1995		
Clause	**Process**	**Clause**	**Process**	**Changes**
7.1.4	Software Detailed Design Process	5.3.6	Software detailed design	Numbering
7.1.5	Software Construction Process	5.3.7	Software coding and testing (renamed Software Construction process by Amd 1)	Title revised; numbering
7.1.6	Software Integration Process	5.3.8	Software integration	Numbering
7.1.7	Software Qualification Testing Process	5.3.9	Software qualification testing	Numbering
7.2	Software Support Processes	6	Supporting life cycle processes	Restructure, Title, Numbering
7.2.1	Software Documentation Management Process	6.1	Documentation process	Title revised; numbering
7.2.2	Software Configuration Management Process	6.2	Configuration management process	Title, Numbering
7.2.3	Software Quality Assurance Process	6.3	Quality assurance process	Title, Numbering
7.2.4	Software Verification Process	6.4	Verification process	Numbering
7.2.5	Software Validation Process	6.5	Validation process	Numbering
7.2.6	Software Review Process	6.6	Joint review process	Numbering
7.2.7	Software Audit Process	6.7	Audit process	Numbering
7.2.8	Software Problem Resolution Process	6.8	Problem resolution process	Numbering
7.3	Software Reuse Processes	7	Organizational life cycle processes	Restructure, title, numbering
7.3.1	Domain Engineering Process	7.7	Domain engineering process	Numbering
7.3.2	Reuse Asset Management Process	7.5	Asset management process	Title; numbering
7.3.3	Reuse Program Management Process	7.6	Reuse program management process	Numbering

Figure 24—Mapping of process clause sets between ISO/IEC 12207:2008 and ISO/IEC 12207:1995

9.2 Relationship description for ISO/IEC 12207:2008 and ISO/IEC 15288:2008

Figure 25 compares the process clause sets between the two revised International Standards. A major change from the prior versions is that there are two main groups of processes in ISO/IEC 12207:2008: those that are software-specific and those that relate to software in the context of its use in a system. The System Context Processes in ISO/IEC 15288:2008 and ISO/IEC 12207:2008 have the same or similar names, structure, purposes and outcomes.

ISO/IEC 12207:2008		ISO/IEC 15288:2008		
Clause	**Process**	**Clause**	**Process**	**Notes**
6	**System life cycle processes**	**6**	**System life cycle processes**	Same title & number
6.1	Agreement processes	6.1	Agreement processes	Same title & number
6.1.1	Acquisition process	6.1.1	Acquisition process	Same title & number
6.1.2	Supply process	6.1.2	Supply process	Same title & number
6.2	Organizational project-enabling processes	6.2	Organizational project-enabling processes	Same title & number
6.2.1	Life cycle model management process	6.2.1	Life cycle model management process	Same title & number
6.2.2	Infrastructure management process	6.2.2	Infrastructure management process	Same title & number
6.2.3	Project portfolio management process	6.2.3	Project portfolio management process	Same title & number
6.2.4	Human resource management process	6.2.4	Human resource management process	Same title & number
6.2.5	Quality management process	6.2.5	Quality management process	Same title & number
6.3	Project processes	6.3	Project processes	Same title & number
6.3.1	Project planning process	6.3.1	Project planning process	Same title & number
6.3.2	Project assessment and control process	6.3.2	Project assessment and control process	Same title & number
6.3.3	Decision management process	6.3.3	Decision management process	Same title & number
6.3.4	Risk management process	6.3.4	Risk management process	Same title & number
6.3.5	Configuration management process	6.3.5	Configuration management process	Same title & number
6.3.6	Information management process	6.3.6	Information management process	Same title & number
6.3.7	Measurement process	6.3.7	Measurement process	Same title & number
6.4	Technical processes	6.4	Technical processes	Same title & number
6.4.1	Stakeholder requirements definition process	6.4.1	Stakeholder requirements definition process	Same title & number
6.4.2	System requirements analysis process	6.4.2	Requirements analysis process	**Similar title; same number**
6.4.3	System architectural design process	6.4.3	Architectural design process	**Similar title; same number**
6.4.4	Implementation process	6.4.4	Implementation process	Same title & number
6.4.5	System integration process	6.4.5	Integration process	**Similar title; same number**

ISO/IEC 12207:2008		ISO/IEC 15288:2008		
6.4.6	System qualification testing process	6.4.6	**Verification process**	**Different topic and title; same number**
6.4.7	Software installation process	6.4.7	**Transition process**	**Different topic and title; same number**
6.4.8	Software acceptance support process	6.4.8	**Validation support process**	**Different topic and title; same number**
6.4.9	Software operation process	6.4.9	operation process	**Similar title; same number**
6.4.10	Software maintenance process	6.4.10	Maintenance process	**Similar title; same number**
6.4.11	Software disposal process	6.4.11	Disposal process	**Similar title; same number**
7	Software life cycle processes			**Clauses for these processes occur only in ISO/IEC 12207:2008. They are not found in ISO/IEC 15288:2008**
7.1	Software implementation processes			
7.1.1	Software implementation process			
7.1.2	Software requirements analysis process			
7.1.3	Software architectural design process			
7.1.4	Software detailed design process			
7.1.5	Software construction process			
7.1.6	Software integration process			
7.1.7	Software qualification testing process			
7.2	Software support processes			
7.2.1	Software documentation management process			
7.2.2	Software configuration management process			
7.2.3	Software quality assurance process			
7.2.4	Software verification process			
7.2.5	Software validation process			
7.2.6	Software review process			
7.2.7	Software audit process			
7.2.8	Software problem resolution process			
7.3	Software reuse process			
7.3.1	Domain engineering process			
7.3.2	Reuse asset management process			
7.3.3	Reuse program management process			

Figure 25—Process comparison between ISO/IEC 12207:2008 and ISO/IEC 15288:2008

Importantly, the process content is yet to be rationalized between the two documents, which is to be addressed in the next revision of the two International Standards. An example of the difference in content of the same process (the Acquisition Process) in one International Standard versus the other is shown in Figure 26. While much of the content of one document does parallel the other, there is content in each that is not in the other and the level of detail is different between the two International Standards.

ISO/IEC 15288:2007	ISO/IEC 12207:2007
6.1.1 Acquisition Process	**6.1.1 Acquisition Process**
6.1.1.1 Purpose	**6.1.1.1 Purpose**
The purpose of the Acquisition Process is to obtain a product or service in accordance with the acquirer's requirements.	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.
6.1.1.2 Outcomes	**6.1.1.2 Outcomes**
As a result of the successful implementation of the Acquisition Process:	As a result of successful implementation of the Acquisition Process:
a) A strategy for the acquisition is established.	a) acquisition needs, goals, product and/or service acceptance criteria and acquisition strategies are defined;
b) One or more suppliers is selected.	b) an agreement is developed that clearly expresses the expectation, responsibilities and liabilities of both the acquirer and the supplier;
c) Communication with the supplier is maintained.	c) one or more suppliers is selected;
d) An agreement to acquire a product or service according to defined acceptance criteria is established.	d) a product and/or service is acquired that satisfies the acquirer’s stated need
e) A product or service complying with the agreement is accepted.	e) the acquisition is monitored so that specified constraints such as cost, schedule and quality are met
f) Payment or other consideration is rendered.	f) supplier deliverables are accepted; and
	g) any identified open items have a satisfactory conclusion as agreed to by the acquirer and the supplier.
6.1.1.3 Activities and Tasks	**6.1.1.3 Activities and Tasks**
The acquirer shall implement the following activities and tasks in accordance with applicable organizational policies and procedures with respect to the Acquisition Process.	The acquirer shall implement the following activities in accordance with applicable organizational policies and procedures with respect to the Acquisition Process.

ISO/IEC 15288:2007	ISO/IEC 12207:2007
NOTE The activities and tasks in this process can apply to one or more suppliers.	NOTE The activities and tasks in this process can apply to one or more suppliers.
a) **Prepare for the Acquisition.** This activity consists of the following tasks:	**6.1.1.3.1 Acquisition Preparation.** This activity consists of the following tasks:
1) Establish a strategy for how the acquisition will be conducted.	**6.1.1.3.1.1** The acquirer begins the acquisition process by describing a concept or a need to acquire, develop, or enhance a system, software product or software service.
NOTE This strategy includes reference to the life cycle model, a schedule of milestones and selection criteria if the supplier is external to the acquiring organization.	
2) Prepare a request for the supply of a product or service that includes the definition of requirements.	**6.1.1.3.1.2** 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.
NOTE Provide a definition of requirements to one or more suppliers. If a supplier is external to organization, then the request can include the business practices with which a supplier is expected to comply and the criteria for selecting a supplier.	
	6.1.1.3.1.3 If the acquirer retains a supplier to perform system requirements analysis, the acquirer shall approve the analyzed requirements.
	6.1.1.3.1.4 The acquirer may perform the definition and analysis of software requirements by itself or may retain a supplier to perform this task.
	6.1.1.3.1.5 The Technical Processes (Clause 6.4) should be used to perform the tasks in clauses 6.1.1.3.1.2 and 6.1.1.3.1.4. The acquirer may use the Stakeholder Requirements Definition Process to establish the customer requirements.
	6.1.1.3.1.6 The acquirer shall consider options for acquisition against analysis of appropriate criteria to include risk, cost and benefits for each option. Options include:
	a) Purchase an off-the-shelf software product that satisfies the requirements.

ISO/IEC 15288:2007	ISO/IEC 12207:2007
	The remaining seven activities under 6.1.1.3.1, Acquisition Preparation, and their associated sub-tasks are not provided as part of this table
Headings only are provided for the rest of the table, below	
b) **Advertise the Acquisition and Select the Supplier.**	**6.1.1.3.2 Acquisition Advertisement.**
c) **Initiate an Agreement.**	**6.1.1.3.3 Supplier Selection**
d) **Monitor the Agreement.**	**6.1.1.3.4 Contract Agreement**
e) **Accept the Product or Service.**	**6.1.1.3.5 Agreement Monitoring**
	6.1.1.3.6 Acquirer Acceptance
	6.1.1.3.7 Closure

Figure 26—Process content difference between ISO/IEC 15288:2008 and ISO/IEC 12207:2008

9.3 General notes on transition

9.3.1 Joint usage of both ISO/IEC 15288 and ISO/IEC 12207

If both standards are used, it is easier for those within the project to use the 2008 versions of both International Standards because of the alignment that has taken place in structure and terminology, facilitating communication, documentation and joint planning of software and system efforts. Converting to the 2008 versions of both standards at the earliest time is recommended, driven by the specific considerations treated under clause 9.4.

9.3.2 Independent usage

If there is no interaction or interdependency between the systems and software effort, the updated version of ISO/IEC 12207 is easier for the software organization to use because of the incorporation of material into the body of the International Standard that was previously in separate amendments. Consequently, conversion to the 2008 version at the earliest time is recommended.

Organizations using only ISO/IEC 15288 may conclude that it is more economical to continue use of the 2002 version. However, there are several fallacies to this position that could be to the commercial disadvantage of the organization. First, with the increasing use of software in systems, failure to anticipate this eventuality would leave the organization caught in a disadvantageous business position when they do have to apply ISO/IEC 12207 for the first time. Either:

a) The organization attempts a hasty, hence risk fraught, conversion to ISO/IEC 15288:2008, or;

b) They could not obtain the business at all because they have not converted while others have, leaving their software-system processes unaligned.

Thus, subject to the specific considerations treated under clause 9.4, earliest possible conversion is still the best business path.

9.4 Notes for new versus existing users

9.4.1 Considerations for transition decisions

Each organization's transition to the updated International Standards must be based on consideration of their unique situation including:

a) Relative system and software content of their projects;

b) Interaction and interdependency between the systems and software parts of the organizations;

c) Investment in procedures, documentation, training and tools for the prior versions;

d) Incremental investment in changing procedures, documentation and tools and in re-training;

e) Stakeholder (especially customer) contractual requirements, preferences, direction, and potential communication problems;

9.4.2 Timing and phasing of transition

An organization can transition to the updated standards on independent or synchronized schedules, with or without a lag in transition on either the system or software International Standard. The transition can be done as one step across an entire organization, or incrementally by project. If the latter, transitioning at project initiation may be better, though the cautions stated in 9.3.2 should be heeded..

9.4.3 Adaptation considerations

Each of the updated International Standards can be adapted as discussed in each standard. As part of that, material from the earlier standard that adds to the updated version is an addition, so full conformance can be maintained while using, permanently or for some duration, material from the earlier version. An example, noted above in clause 9.3.2, would be adapting ISO/IEC 15288:2008 by adding material from the 2002 version.

The approach can also be used if, after carefully considering clause 9.3, gating factors such as stated in clauses 9.4.1 and 9.4.2 force continue use of ISO/IEC 15288:2002. That is, parts of ISO/IEC 15288:2008 could be selected that add to the 2002 material. So, if a specific process had more detail under the new (2008) version, the additional detail could be employed to adapt the 2002 version additively.

These are important options to think through for the organization's specific needs as it allows use of either version as a baseline, while taking advantage of either the prior version's or the new version's material that is helpful to the organization.

9.5 Notes on using application guides ISO/IEC TR 15271 and ISO/IEC TR 19760

It requires some thought about specific details for people to use the application guides written for the earlier versions of the two International Standards. However, their general themes and content nonetheless gives a useful guide to applying the two updated International Standards. Their use, with prudent care, is feasible until updates to or replacement of the application guides is completed.

9.6 Adjustments in relationships with other ISO and ISO/IEC documents

ISO and ISO/IEC documents have evolved and this will continue. It would be useful if the nature and timing of the changes could be forecast at a usable level of detail, but that is not possible. What is clear is that the documents and their application guides will increasingly align with each other over an extended span of time, consequently it is in the user's interest to understand how to make best use of the standard that is in hand yet maintain maximum flexibility to change as updates come along, or accept that this cannot be done and periodic change and investment is inevitable.

9.7 Developing a forward strategy

Each organization's forward strategy in handling the evolution of a set of interrelating International Standards is unique and rests on the same factors stated in clause 9.4.1. Beyond that, the best element of strategy is to invest in maintaining awareness of what standards are being addressed and what changes are surfacing as revision drafts move through each stage. This allows the organization to project possible impacts and investment needs several years into the future, refining the projections into specific plans as updates near publication. While some reinvestment may remain necessary, including this one element in the overall strategy provides the forward visibility to reduce the reinvestment compared to what otherwise might be necessary.

10 Bibliography

ISO/IEC 12207:2008 Systems and software engineering—Software life cycle processes

ISO 13407:1999, Human-centered design processes for interactive systems

ISO/IEC TR 15271:1998 Information Technology—Guide for ISO/IEC 12207

ISO/IEC 15288:2008 Systems and software engineering—System life cycle processes

ISO/IEC 15939:2007 Systems and software engineering—Measurement process

ISO/IEC 16085:2006 Systems and software engineering -- Life cycle processes -- Risk management

ISO/TR 18529:2000 Ergonomics—Ergonomics of human-system interaction—Human-centered lifecycle process descriptions

ISO/IEC TR 19760:2003 Systems Engineering—Guide for the application of ISO/IEC 15288

ISO/IEC 26702:2007 Systems engineering -- Application and management of the systems engineering process:

ISO 9000:2005 Quality management systems—Fundamentals and vocabulary

IEC 61150 (2006), Design Review

IEEE Std 1028-1997, IEEE Standard for Software Reviews

IEEE Std 1044-1993, IEEE Standard Classification for Software Anomalies

Annex A

Terms and definitions from other sources used in ISO/IEC TR 24748

For the purposes of this Technical Report, the following terms and definitions, common across ISO/IEC 12207, ISO/IEC 15288 and this document, apply.

A.1
acquirer
the stakeholder that acquires or procures a product or service from a supplier.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

NOTE The acquirer could be one of the following: buyer, customer, owner, or purchaser.

A.2
acquisition
the process of obtaining a product or service.

[ISO/IEC 15288:2008]

A.3
activity
a set of cohesive tasks of a process.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.4
agreement
the mutual acknowledgement of terms and conditions under which a working relationship is conducted.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.5
architecture
the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.

[ISO/IEC 42010:2008]

NOTE 1 ISO/IEC 12207 and ISO/IEC 15288 use the word elements instead of components and this Technical Report follows that usage.

A.6
audit
systematic, independent and documented process for obtaining audit evidence and evaluating it objectively to determine the extent to which audit criteria are fulfilled.

[ISO 9000:2005]

A.7
baseline
a specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures.

[ISO/IEC 15288:2008]

A.8
contract
a binding agreement between two parties, especially enforceable by law, or a similar internal agreement wholly within an organization.

[ISO/IEC 12207:2008]

A.9
customer
organization or person that receives a product or service

NOTE 1 A customer can be internal or external to the organization.

[Adapted from ISO 9000:2005]

NOTE 2 Other terms commonly used for customer are acquirer, buyer, or purchaser.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.10
developer
an organization that performs development activities (including requirements analysis, design, testing through acceptance) during the life cycle process.

[Adapted from ISO/IEC 12207:2008]

A.11
enabling system
a system that supports a system-of-interest during its life cycle stages but does not necessarily contribute directly to its function during operation.

[Adapted from ISO/IEC 15288:2008]

NOTE 1 For example, when a system-of-interest enters the production stage, a production enabling system is required.

NOTE 2 Each enabling system has a life cycle of its own. This Technical Report is applicable to each enabling system when, in its own right, it is treated as a system-of-interest.

A.12
evaluation
a systematic determination of the extent to which an entity meets its specified criteria.

[ISO/IEC 12207:2008]

A.13
facility
the physical means or equipment for facilitating the performance of an action, e.g. buildings, instruments, tools.

[ISO/IEC 15288:2008]

A.14
life cycle
the evolution of a system, product, service, project or other human-made entity from conception through retirement.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.15
life cycle model
a framework of processes and activities (which may be organized into stages) concerned with the life cycle, which also acts as a common reference for communication and understanding.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.16
maintainer
an individual who, or organization that, performs maintenance activities.

[Adapted from ISO/IEC 12207:2008]

A.17
monitoring
an examination of the status of the activities of a supplier and of their results by the acquirer or a third party.

[ISO/IEC 12207:2008]

A.18
operator
the entity that performs the operation of a system.

[ISO/IEC 15288:2008]

NOTE 1 The role of operator and the role of user may be vested, simultaneously or sequentially, in the same individual or organization.

NOTE 2 An individual operator combined with knowledge, skills and procedures may be considered as an element of the system.

NOTE 3 In the context of this specific definition, the term entity means an individual or an organization.

A.19
organization
a person or a group of people and facilities with an arrangement of responsibilities, authorities and relationships.

[Adapted from ISO 9000:2005]

NOTE 1 A body of persons organized for some specific purpose, such as a club, union, corporation, or society, is an organization.

NOTE 2 An identified part of an organization (even as small as a single individual) or an identified group of organizations can be regarded as an organization if it has responsibilities, authorities and relationships.

A.20
party
an organization entering into an agreement

NOTE In this Technical Report, the agreeing parties are called the acquirer and the supplier.

[ISO/IEC 15288:2008]

A.21
process
a set of interrelated or interacting activities which transforms inputs into outputs.

[ISO 9000:2005]

A.22
process purpose
the high level objective of performing the process and the likely outcomes of effective implementation of the process. The implementation of the process should provide tangible benefits to the stakeholders

[ISO/IEC 12207:2008]

A.23
process outcome
an observable result of the successful achievement of the process purpose

[ISO/IEC 12207:2008]

A.24
product
the result of a process

[ISO 9000:2005]

A.25
project
an endeavour with defined start and finish criteria undertaken to create a product or service in accordance with specified resources and requirements.

[Adapted from ISO 9000: 2005]

NOTE 1 A project may be viewed as a unique process comprising co-ordinated and controlled activities and may be composed of activities from the Project Processes and Technical Processes referred to in this Technical Report.

NOTE 2 A process can also be viewed as a specific instantiation of lifecycle processes, adapted within a life cycle model, to create the service or product for the specific requirements and context of the project.

A.26
qualification
the process of demonstrating whether an entity is capable of fulfilling specified requirements.

[ISO/IEC 12207:2008]

A.27
quality assurance
part of quality management focused on providing confidence that quality requirements will be fulfilled

[ISO 9000: 2005]

A.28
resource
an asset that is utilized or consumed during the execution of a process.

[ISO/IEC 15288:2008]

NOTE 1 Resources may include diverse entities such as funds, personnel, facilities, capital equipment, tools, and utilities such as power, water, fuel and communication infrastructures.

NOTE 2 Resources may be reusable, renewable or consumable.

A.29
retirement
withdrawal of active support by the operation and maintenance organization, partial or total replacement by a new system, or installation of an upgraded system.

[ISO/IEC 12207:2008]

A.30
security
all aspects related to defining, achieving, and maintaining confidentiality, integrity, availability, non-repudiation, accountability and authenticity of a system.

[Adapted from ISO/IEC 13335-1: 2004]

A.31
service
performance of activities, work, or duties associated with a product.

[ISO/IEC 12207:2008]

A.32
software product
the set of computer programs, procedures, and possibly associated documentation and data.

[ISO/IEC 12207:2008]

A.33
stage
a period within the life cycle of an entity that relates to the state of its description or realization.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

NOTE 1 As used in this Technical Report, stages relate to major progress and achievement milestones of the entity through its life cycle.

NOTE 2 Stages may be overlapping.

A.34
stakeholder
an individual or organization having a right, share, claim, or interest in a system or in its possession of characteristics that meet their needs and expectations.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.35
supplier
an organization or an individual that enters into an agreement with the acquirer for the supply of a product or service.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

NOTE 1 Other terms commonly used for supplier are contractor, producer, seller, or vendor.

NOTE 2 The acquirer and the supplier may be part of the same organization.

A.36
system
a combination of interacting elements organized to achieve one or more stated purposes.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

NOTE 1 A system may be considered as a product or as the services it provides.

NOTE 2 In practice, the interpretation of its meaning is frequently clarified by the use of an associative noun, e.g. aircraft system. Alternatively the word "system" may be substituted simply by a context-dependent synonym, e.g. aircraft, though this may then obscure a system principles perspective.

A.37
system element
a member of a set of elements that constitutes a system.

[ISO/IEC 15288:2008]

NOTE A system element is a discrete part of a system that can be implemented to fulfil specified requirements. A system element can be hardware, software, data, humans, processes (e.g. processes for providing service to users), procedures (e.g. operator instructions), facilities, materials, and naturally occurring entities (e.g. water, organisms, minerals), or any combination.

A.38
system-of-interest
the system whose life cycle is under consideration in the context of this Technical Report.

[ISO/IEC 15288:2008]

A.39
task
a requirement, recommendation, or permissible action, intended to contribute to the achievement of one or more outcomes of a process.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

A.40
trade-off
decision-making actions that select from various requirements and alternative solutions on the basis of net benefit to the stakeholders.

[ISO/IEC 15288:2008]

A.41
user
individual or group that benefits from a system during its utilization.

[ISO/IEC 12207:2008 and ISO/IEC 15288:2008]

NOTE The role of user and the role of operator may be vested, simultaneously or sequentially, in the same individual or organization.

A.42
validation
confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled.

[ISO 9000:2005]

NOTE Validation is the set of activities ensuring and gaining confidence that a system is able to accomplish its intended use, goals and objectives (i.e. meet stakeholder requirements) in the intended operational environment.

A.43
verification
confirmation, through the provision of objective evidence, that specified requirements have been fulfilled.

[ISO 9000:2005]

NOTE Verification is a set of activities that compares a product of the life cycle against the required characteristics for that product. This may include, but is not limited to, specified requirements, design description and the system itself.

A.44
version
an identified instance of an item.

[ISO/IEC 12207:2008]

NOTE Modification to a version of a product, resulting in a new version, requires configuration management action.

Annex B

Guidance on development strategies and build planning

B.1 Scope

This annex describes three candidate development strategies.

B.2 Candidate development strategies

There are many different development strategies that can be applied to system and software projects. Three of these strategies are summarized below:

a) Once-through. The "once-through" strategy, also called "waterfall," consists of performing the development process a single time. Simplistically: determine user needs, define requirements, design the system, implement the system, test, fix, and deliver.

b) Incremental. The "incremental" strategy determines user needs and defines the system requirements, then performs the rest of the development in a sequence of builds. The first build incorporates part of the planned capabilities, the next build adds more capabilities, and so on, until the system is complete.

c) Evolutionary. The "evolutionary" strategy also develops a system in builds but differs from the incremental strategy in acknowledging that the user need is not fully understood and all requirements cannot be defined up front. In this strategy, user needs and system requirements are partially defined up front, and then are refined in each succeeding build.

Characteristics of the three strategies are summarized in Figure B.1 below:

Development strategy	Define all requirements first?	Multiple development cycles?	Provide interim function?
Once-through (Waterfall)	**Yes**	**No**	**No**
Incremental (Preplanned Product Improvement)	**Yes**	**Yes**	**Maybe**
Evolutionary	**No**	**Yes**	**Yes**

Figure B.1 Summary of strategy characteristics

B.3 Selecting an appropriate development strategy

The development strategy should be agreed on by the acquirer and the supplier. The acquirer should carefully state any program-specific requirements that may affect the selection of a development strategy (e.g., funding profile limitations, early prototype demonstrations, evolving operational requirements). The supplier should propose, subject to acquirer approval, the development strategy that most effectively satisfies the program requirements. A specific strategy should be required by the acquirer only when the acquirer has a complete understanding of the programmatic and technical domains and determines that no other approach will be entertained.

The Figure B.2, following, illustrates a sample risk analysis approach for selecting an appropriate strategy. The approach consists of listing risk items (negatives) and opportunity items (positives) for each strategy; assigning each item a risk or opportunity level of High, Medium, or Low; and making a decision on which strategy to use based on a trade-off among the risks and opportunities. The fill-ins shown are sample considerations only. An actual analysis may use others. The "DECISION" entry on the bottom line shows which strategy was selected for this particular illustration.

Once-Through		Incremental		Evolutionary	
Risk (Reasons against this strategy)	**Risk Level**	**Risk (Reasons against this strategy)**	**Risk Level**	**Risk (Reasons against this strategy)**	**Risk Level**
Requirements are not well understood	H	Requirements are not well understood	H		
System too large to do all at once	M	User prefers all capabilities at first delivery	M	User prefers all capabilities at first delivery	M
Rapid changes in technology anticipated—may change the requirements	H	Rapid changes in technology are expected—may change the requirements	H		
Limited staff or budget available now	M				
Opportunity (Reasons to use this strategy)	**Opp. Level**	**Opportunity (Reasons to use this strategy)**	**Opp. Level**	**Opportunity (Reasons to use this strategy)**	**Opp. Level**
User prefers all capabilities at first delivery	M	Early capability is needed	H	Early capability is needed	H
User prefers to phase out old system all at once	L	System breaks naturally into increments	M	System breaks naturally into increments	M
		Funding/staffing will be incremental	H	Funding/staffing will be incremental	H
				User feedback and monitoring of technology changes is needed to understand full requirements	H
				DECISION: USE THIS STRATEGY	

Figure B.2 Illustration of strategy selection

B.4 Relationship of systems and software to development strategies

The development strategy usually applies to the overall system. The software within the system may be acquired under the same strategy or under a different one, such as requiring that all software be finalized in the first build of the system.

B.5 Planning software builds

Planning the software builds on a project may be accomplished in several ways. The acquirer might, for example, select an overall development strategy, leaving it to the supplier to lay out the software builds. Alternatively, the acquirer might lay out the software builds as part of the contract. The approach selected will be project-dependent. Of course, the software build plan must support the plan for incremental delivery of system function. The sub-clauses below provide guidelines for planning the builds without attempting to identify who performs each of the activities.

B.5.1 Identifying builds and their objectives

The first step in software build planning is to lay out a series of one or more builds and to identify the objectives of each build.

B.5.2 Identifying the activities to be performed in each build

The next step in build planning is identifying which activities apply in each build and determining the extent to which they apply. Some activities may not apply at all in a given build, some may apply identically in all builds, and some may apply differently in different builds. The following guidelines apply:

a) Different decisions may apply to different types of software on a project.

b) If early builds are devoted to experimentation, developing “throw-away” software to arrive at a system concept or system requirements, it may be appropriate to forgo certain formalities, such as coding standards, that may be imposed later on the “real” software. If the early software is to be used later, such formalities may be appropriate from the start. These decisions are project-dependent.

B.5.3 Recording build planning decisions

Build planning decisions made by the acquirer before the project begins are specified in the contract. Build planning proposed by the supplier may be communicated via feedback on draft requests for proposal, proposals, the software plan, joint reviews during the project, or by other means of communication. Refinements to the build planning decisions may be ongoing as the project proceeds. Those decisions involving contractual changes are handled accordingly.

B.5.4 Scheduling the selected activities in each build

Another important step in build planning is scheduling the activities in each build. The acquirer may set forth general milestones and have the supplier provide specifics or may provide specific schedules. The following guidelines apply:

a) A common mistake is to treat all software items as though they are required to be developed in “lock-step,” reaching key milestones at the same time. Allowing software items to be on different schedules can result in more optimum development.

b) A similar mistake is to treat software units as though they are required to be developed in “lock-step,” all designed by a certain date, implemented by a certain date, etc. Flexibility in the scheduling of software units can also be effective.

c) The activities in the standard need not be performed sequentially. Several may be taking place at one time, and an activity may be performed continually or intermittently throughout a build or over multiple builds. The activities in each build are to be laid out in the manner that best suits the work to be done.

Annex C

Candidate joint management reviews

C.1 Scope

In two-party acquisition situations, there is often a need for joint management reviews between the two parties to the agreement. Often these joint reviews contribute to the decision gates marking the milestones between project stages. This annex describes a candidate set of joint management reviews that might be held during a project.

C.2 Assumptions

This annex makes the following assumptions:

a) The acquirer has reviewed the subject products in advance, and one or more joint technical reviews have been held to resolve issues, leaving the joint management review as a forum to resolve open issues and reach agreement as to the acceptability of each product;

b) Any of the reviews may be conducted incrementally, dealing at each review with a subset of the listed items or a subset of the system or software items being reviewed.

C.3 Candidate reviews

Given below is a set of candidate joint management reviews that might be held during a project. There is no intent to require these reviews or to preclude alternatives or combinations of these reviews. All products are expressed in lower case letters to emphasize that they are generic in nature. Furthermore, there is no intent to suggest that documentation products are in printed form, nor that they exist in some unitary form.

C.3.1 Plan reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) Project plans;

b) Test plans;

c) Development, test, operational, maintenance environment and tools.

C.3.2 Operational concept reviews

These reviews are held to resolve open issues regarding the operational concept for a system.

C.3.3 System requirements reviews

These reviews are held to resolve open issues regarding the specified requirements for a system or element.

C.3.4 System design reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) The system design decisions;

b) The architectural design of a system or its elements.

C.3.5 Software requirements reviews

These reviews are held to resolve open issues regarding the requirements derived for a software item.

C.3.6 Software design reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) The software-wide design decisions;

b) The architectural design of a software item;

c) The detailed design of a software item or portion thereof (such as a database).

C.3.7 Test readiness reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) The status of the test environment;

b) The test cases and test procedures to be used for testing;

c) The status of the system or element to be tested.

C.3.8 Test results reviews

These reviews are held to resolve open issues regarding the results of testing.

C.3.9 Usability reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) The readiness of the product for installation at user sites;

b) Status of training;

c) The user and operator manuals;

c) The version descriptions;

e) The status of installation preparations and activities.

C.3.10 Maintenance reviews

These reviews are held to resolve open issues regarding one or more of the following:

a) The readiness of the product for transition to the maintenance organization;

b) The product specifications;

c) The maintenance manuals;

d) The version descriptions;

e) The status of transition preparations and activities, including transition of the engineering environment, if applicable.

C.3.11 Critical requirement reviews

These reviews are held to resolve open issues regarding the handling of critical requirements, such as those for safety, security, and privacy protection.

C.4 Other resources

Additional information on joint reviews may be found in IEEE Std 1028-1997, IEEE Standard for Software Reviews, and in IEC 61150 (2006), Design Review.

Annex D

Problem reporting capability

D.1 Unified problem reporting

Whichever model is used, in dealing with problems arising during the execution of life cycle processes, it is useful to maintain a unified problem reporting capability. By its nature, a unified capability may deal with large numbers of problems. It is therefore useful to categorize the problems in various ways. This clause recommends two classifications: by category and; by priority. Others may be applicable in particular situations.

D.2 Problem classification

Figure D.1 shows problem classification by category and Figure D.2 shows classification by priority, both for software, though these can readily be generalized. Any given problem can be assigned to more than one of the categories in Figure D.1 if classification is by category, while only one priority can be assigned if classification is done by priority, following Figure D.2. IEEE Std 1044-1993, *IEEE Standard Classification for Software Anomalies*, may be helpful in developing more detailed schemes for classifying software-related anomalies.

Category	Applies to problems in
a. Plans	One of the plans developed for the project
b. Concept	The operational concept
c. Requirements	The system or software requirements
d. Design	The design of the system or software
e. Code	The software code
f. Data	A database or data file
g. Test information	Test plans, test descriptions, or test reports
h. Manuals	The user, operator, or maintenance manuals
i. Other	Other hardware or software products

Figure D.1 Problem classification by category

Priority	If a problem could
1	a) Prevent the accomplishment of an essential capability, or b) Jeopardize safety, security, or other requirement designated “critical”
2	a) Adversely affect the accomplishment of an essential capability and no work-around solution is known, or b) Adversely affect technical, cost, or schedule risks to the project or to life cycle support of the system, and no work-around solution is known
3	a) Adversely affect the accomplishment of an essential capability but a work-around solution is known, or b) Adversely affect technical, cost, or schedule risks to the project or to life cycle support of the system, but a work-around solution is known
4	a) Result in user/operator inconvenience or annoyance but does not affect a required operational or mission-essential capability, or b) Result in inconvenience or annoyance for development or maintenance personnel but does not prevent the accomplishment of the responsibilities of those personnel
5	Have any other effect

Figure D.2 Problem classification by priority