

INTERNATIONAL
STANDARD

ISO/IEC
12207

First edition
1995-08-01

**Information technology — Software life
cycle processes**

iTeh STANDARD PREVIEW

Technologies de l'information — Processus du cycle de vie des logiciels
(standards.iteh.ai)

[ISO/IEC 12207:1995](#)

[https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-
bf9047d3ffb/iso-iec-12207-1995](https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-bf9047d3ffb/iso-iec-12207-1995)



Reference number
ISO/IEC 12207:1995(E)

Contents

Foreword	iii
Introduction	iv
1 Scope	1
2 Normative references	2
3 Definitions	3
4 Application of this International Standard	6
5 Primary life cycle processes	9
5.1 Acquisition process	10
5.2 Supply process	13
5.3 Development process	16
5.4 Operation process	23
5.5 Maintenance process	24
6 Supporting life cycle processes	27
6.1 Documentation process	28
6.2 Configuration management process	29
6.3 Quality assurance process	31
6.4 Verification process	33
6.5 Validation process	36
6.6 Joint review process	38
6.7 Audit process	40
6.8 Problem resolution process	41
7 Organizational life cycle processes	42
7.1 Management process	43
7.2 Infrastructure process	45
7.3 Improvement process	46
7.4 Training process	47
Annexes	
A - Tailoring process	48
B - Guidance on tailoring	49
C - Guidance on processes and organizations	53
D - Bibliography	57

Iteh STANDARD PREVIEW
 (standards.iteh.ai)
 ISO/IEC 12207:1995
<https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-41250-b9047d3ffb/iso-iec-12207-1995>

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. 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.

International Standard ISO/IEC 12207 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software engineering*.

Annex A forms an integral part of this International Standard. Annexes B and C are for information only.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 12207:1995](https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995)

<https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995>

Introduction

Software is an integral part of information technology and conventional systems, such as transportation, military, medical care, and finance. There is a proliferation of standards, procedures, methods, tools, and environments for developing and managing software. This proliferation has created difficulties in software management and engineering, especially in integrating products and services. The software discipline needs to migrate from this proliferation to a common framework that can be used by software practitioners to "speak the same language" to create and manage software. This International Standard provides such a common framework.

The framework covers the life cycle of software from conceptualization of ideas through retirement and consists of processes for acquiring and supplying software products and services. In addition, the framework provides for controlling and improving these processes.

The processes in this International Standard form a comprehensive set. An organization, depending on its purpose, can select an appropriate subset to fulfill that purpose. This International Standard is, therefore, designed to be tailored for an individual organization, project, or application. It is also designed to be used when software is a stand-alone entity, or an embedded or integral part of the total system.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 12207:1995](https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995)

<https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995>

Information technology – Software life cycle processes

1 Scope

1.1 Purpose

This International Standard establishes a common framework for software life cycle processes, with well-defined terminology, that can be referenced by the software industry. It contains processes, activities, and tasks that are to be applied during the acquisition of a system that contains software, a stand-alone software product, and software service and during the supply, development, operation, and maintenance of software products. Software includes the software portion of firmware.

This International Standard also provides a process that can be employed for defining, controlling, and improving software life cycle processes.

1.2 Field of application

This International Standard applies to the acquisition of systems and software products and services, to the supply, development, operation, and maintenance of software products, and to the software portion of firmware, whether performed internally or externally to an organization. Those aspects of system definition needed to provide the context for software products and services are included.

NOTE – The processes used during the software life cycle need to be compatible with the processes used during the system life cycle.

This International Standard is intended for use in a two-party situation and may be equally applied where the two parties are from the same organization. The situation may range from an informal agreement up to a legally binding contract. This International Standard may be used by a single party as self-imposed tasks.

This International Standard is not intended for off-the-shelf software products unless incorporated into a deliverable product.

This International Standard is written for acquirers of systems and software products and services and for suppliers, developers, operators, maintainers, managers, quality assurance managers, and users of software products.

1.3 Tailoring of this International Standard

This International Standard contains a set of processes, activities, and tasks designed to be tailored in respect of software projects. The tailoring process is deletion of non-applicable processes, activities, and tasks.

NOTE – Addition of unique or special processes, activities, and tasks may be provided in the contract.

1.4 Compliance

Compliance with this International Standard is defined as the performance of all the processes, activities, and tasks selected from this International Standard in the Tailoring Process (annex A) for the software project. The performance of a process or an activity is complete when all its required tasks are performed in accordance with the pre-established criteria and the requirements specified in the contract as applicable.

Any organization (for example, national, industrial association, company) imposing this International Standard, as a condition of trade, is responsible for specifying and making public the minimum set of required processes, activities, and tasks, which constitute suppliers' compliance with this International Standard.

1.5 Limitations

This International Standard describes the architecture of the software life cycle processes but does not specify the details of how to implement or perform the activities and tasks included in the processes.

This International Standard is not intended to prescribe the name, format, or explicit content of the documentation to be produced. This International Standard may require development of documents of similar class or type; various plans are an example. This International Standard, however, does not imply that such documents be developed or packaged separately or combined in some fashion. These decisions are left to the user of this International Standard.

This International Standard does not prescribe a specific life cycle model or software development method. The parties of this International Standard are responsible for selecting a life cycle model for the software project and mapping the processes, activities, and tasks in this International Standard onto that model. The parties are also responsible for selecting and applying the software development methods and for performing the activities and tasks suitable for the software project.

This International Standard is not intended to be in conflict with any organization's policies, standards or procedures that are already in place. However, any conflict needs to be resolved and any overriding conditions and situations need to be cited in writing as exceptions to the application of this International Standard.

Throughout this International Standard, "shall" is used to express a provision that is binding between two or more parties, "will" to express a declaration of purpose or intent by one party, "should" to express a recommendation among other possibilities, and "may" to indicate a course of action permissible within the limits of this International Standard.

In this International Standard, there are a number of lists for tasks; none of these is presumed to be exhaustive -- they are intended as examples.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/AFNOR: 1989, *Dictionary of computer science*.

ISO/IEC 2382-1: 1993, *Information technology – Vocabulary – Part 1: Fundamental terms*.

ISO/IEC 2382-20: 1990, *Information technology – Vocabulary – Part 20: System development*.

ISO 8402: 1994, *Quality management and quality assurance – Vocabulary*.

ISO 9001: 1994, *Quality systems – Model for quality assurance in design, development, production, installation and servicing.*

ISO/IEC 9126: 1991, *Information technology – Software product evaluation – Quality characteristics and guidelines for their use.*

3 Definitions

For the purposes of this International Standard, the definitions given in ISO 8402, ISO/IEC 2382-1 and ISO/IEC 2382-20 apply, together with the following definitions.

NOTE – A product may be interpreted as a part of a system as applicable.

3.1 Acquirer: An organization that acquires or procures a system, software product or software service from a supplier.

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

3.2 Acquisition: The process of obtaining a system, software product or software service.

3.3 Agreement: The definition of terms and conditions under which a working relationship will be conducted.

3.4 Audit: Conducted by an authorized person for the purpose of providing an independent assessment of software products and processes in order to assess compliance with requirements.

3.5 Baseline: A formally approved version of a configuration item, regardless of media, formally designated and fixed at a specific time during the configuration item's life cycle.

3.6 Configuration item: An entity within a configuration that satisfies an end use function and that can be uniquely identified at a given reference point.

3.7 Contract: A binding agreement between two parties, especially enforceable by law, or a similar internal agreement wholly within an organization, for the supply of software service or for the supply, development, production, operation, or maintenance of a software product.

3.8 Developer: An organization that performs development activities (including requirements analysis, design, testing through acceptance) during the software life cycle process.

3.9 Evaluation: A systematic determination of the extent to which an entity meets its specified criteria.

3.10 Firmware: The combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control.

3.11 Life cycle model: A framework containing the processes, activities, and tasks involved in the development, operation, and maintenance of a software product, spanning the life of the system from the definition of its requirements to the termination of its use.

3.12 Maintainer: An organization that performs maintenance activities.

3.13 Monitoring: An examination of the status of the activities of a supplier and of their results by the acquirer or a third party.

3.14 Non-deliverable item: Hardware or software product that is not required to be delivered under the contract but may be employed in the development of a software product.

3.15 Off-the-shelf product: Product that is already developed and available, usable either "as is" or with modification.

3.16 Operator: An organization that operates the system.

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

NOTE – The term "activities" covers use of resources. [See ISO 8402: 1994, 1.2.]

3.18 Qualification: The process of demonstrating whether an entity is capable of fulfilling specified requirements. [See ISO 8402: 1994, 2.13.]

3.19 Qualification requirement: A set of criteria or conditions that have to be met in order to qualify a software product as complying with its specifications and being ready for use in its target environment.

3.20 Qualification testing: Testing, conducted by the developer and witnessed by the acquirer (as appropriate), to demonstrate that a software product meets its specifications and is ready for use in its target environment.

3.21 Quality assurance: All the planned and systematic activities implemented within the quality system, and demonstrated as needed, to provide adequate confidence that an entity will fulfil requirements for quality.

NOTES

1 There are both internal and external purposes for quality assurance:

- a) Internal quality assurance: within an organization, quality assurance provides confidence to management;
- b) External quality assurance: in contractual situations, quality assurance provides confidence to the customer or others.

2 Some quality control and quality assurance actions are interrelated.

3 Unless requirements for quality fully reflect the needs of the user, quality assurance may not provide adequate confidence.

[ISO 8402: 1994, 3.5]

3.22 Release: A particular version of a configuration item that is made available for a specific purpose (for example, test release).

3.23 Request for proposal [tender]: A document used by the acquirer as the means to announce its intention to potential bidders to acquire a specified system, software product or software service.

3.24 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.

3.25 Security: The protection of information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or systems are not denied access to them.

3.26 Software product: The set of computer programs, procedures, and possibly associated documentation and data.

3.27 Software service: Performance of activities, work, or duties connected with a software product, such as its development, maintenance, and operation.

3.28 Software unit: A separately compilable piece of code.

3.29 Statement of work: A document used by the acquirer as the means to describe and specify the tasks to be performed under the contract.

3.30 Supplier: An organization that enters into a contract with the acquirer for the supply of a system, software product or software service under the terms of the contract.

NOTES

1 The term "supplier" is synonymous with contractor, producer, seller, or vendor.

2 The acquirer may designate a part of its organization as supplier.

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

3.32 Test coverage: The extent to which the test cases test the requirements for the system or software product.

3.33 Testability: The extent to which an objective and feasible test can be designed to determine whether a requirement is met.

3.34 User: An individual or organization that uses the operational system to perform a specific function.

NOTE – The user may perform other roles, such as acquirer, developer, or maintainer.

3.35 Validation: Confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled.

NOTES

1 In design and development, validation concerns the process of examining a product to determine conformity with user needs.

2 Validation is normally performed on the final product under defined operating conditions. It may be necessary in earlier stages.

3 "Validated" is used to designate the corresponding status.

4 Multiple validations may be carried out if there are different intended uses.

[ISO 8402: 1994, 2.18]

3.36 Verification: Confirmation by examination and provision of objective evidence that specified requirements have been fulfilled.

NOTES

1 In design and development, verification concerns the process of examining the result of a given activity to determine conformity with the stated requirement for that activity.

2 "Verified" is used to designate the corresponding status.

[ISO 8402: 1994, 2.17]

3.37 Version: An identified instance of an item.

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

4 Application of this International Standard

This clause presents the software life cycle processes that can be employed to acquire, supply, develop, operate, and maintain software products. The objective is to provide a road map for the users of this International Standard so that they can orient themselves in it and apply it judiciously.

4.1 Organization of this International Standard

4.1.1 Life cycle processes

This International Standard groups the activities that may be performed during the life cycle of software into five primary processes, eight supporting processes, and four organizational processes. Each life cycle process is divided into a set of activities; each activity is further divided into a set of tasks. Subclause numbering a.b denotes a process, a.b.c an activity, and a.b.c.d a task. These life cycle processes are introduced below and depicted in figure 1.

4.1.1.1 Primary life cycle processes

The primary life cycle processes (clause 5) consist of five processes that serve primary parties during the life cycle of software. A primary party is one that initiates or performs the development, operation, or maintenance of software products. These primary parties are the acquirer, the supplier, the developer, the operator, and the maintainer of software products. The primary processes are:

- 1) *Acquisition process* (subclause 5.1). Defines the activities of the acquirer, the organization that acquires a system, software product or software service.
- 2) *Supply process* (subclause 5.2). Defines the activities of the supplier, the organization that provides the system, software product or software service to the acquirer.
- 3) *Development process* (subclause 5.3). Defines the activities of the developer, the organization that defines and develops the software product.
- 4) *Operation process* (subclause 5.4). Defines the activities of the operator, the organization that provides the service of operating a computer system in its live environment for its users.
- 5) *Maintenance process* (subclause 5.5). Defines the activities of the maintainer, the organization that provides the service of maintaining the software product; that is, managing modifications to the software product to keep it current and in operational fitness. This process includes the migration and retirement of the software product.

4.1.1.2 Supporting life cycle processes

The supporting life cycle processes (clause 6) consist of eight processes. A supporting process supports another process as an integral part with a distinct purpose and contributes to the success and quality of the software project. A supporting process is employed and executed, as needed, by another process. The supporting processes are:

- 1) *Documentation process* (subclause 6.1). Defines the activities for recording the information produced by a life cycle process.
- 2) *Configuration management process* (subclause 6.2). Defines the configuration management activities.
- 3) *Quality assurance process* (subclause 6.3). Defines the activities for objectively assuring that the software products and processes are in conformance with their specified requirements and adhere to their established plans. Joint Reviews, Audits, Verification, and Validation may be used as techniques of Quality Assurance.

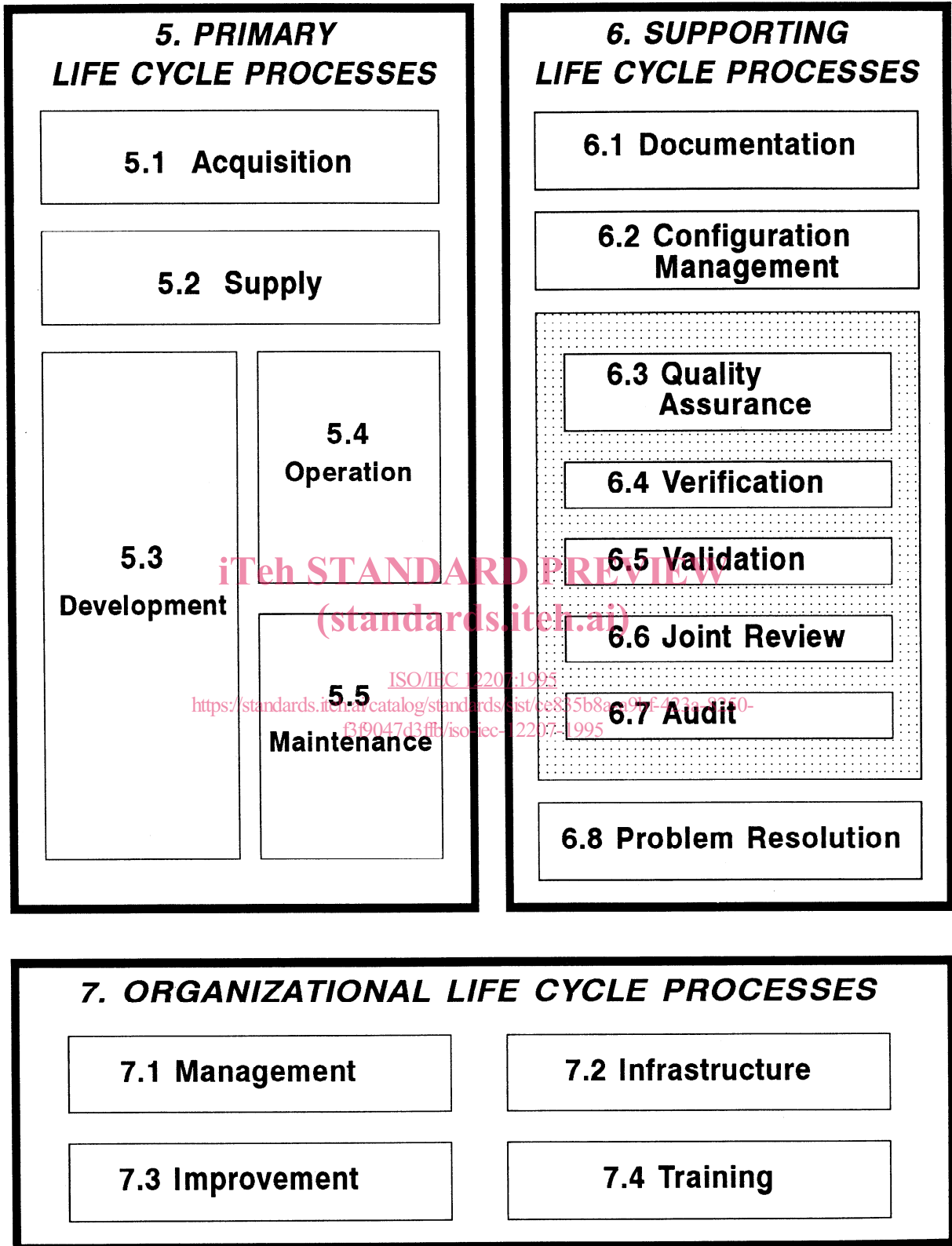


Figure 1. Structure of the International Standard

- 4) *Verification process* (subclause 6.4). Defines the activities (for the acquirer, the supplier, or an independent party) for verifying the software products in varying depth depending on the software project.
- 5) *Validation process* (subclause 6.5). Defines the activities (for the acquirer, the supplier, or an independent party) for validating the software products of the software project.
- 6) *Joint review process* (subclause 6.6). Defines the activities for evaluating the status and products of an activity. This process may be employed by any two parties, where one party (reviewing party) reviews another party (reviewed party) in a joint forum.
- 7) *Audit process* (subclause 6.7). Defines the activities for determining compliance with the requirements, plans and contract. This process may be employed by any two parties, where one party (auditing party) audits the software products or activities of another party (audited party).
- 8) *Problem resolution process* (subclause 6.8). Defines a process for analyzing and removing the problems (including non-conformances), whatever their nature or source, that are discovered during the execution of development, operation, maintenance, or other processes.

4.1.1.3 Organizational life cycle processes

The organizational life cycle processes (clause 7) consist of four processes. They are employed by an organization to establish and implement an underlying structure made up of associated life cycle processes and personnel and continuously improve the structure and processes. They are typically employed outside the realm of specific projects and contracts; however, lessons from such projects and contracts contribute to the improvement of the organization. The organizational processes are:

- 1) *Management process* (subclause 7.1). Defines the basic activities of the management, including project management, during a life cycle process.
- 2) *Infrastructure process* (subclause 7.2). Defines the basic activities for establishing the underlying structure of a life cycle process.
- 3) *Improvement process* (subclause 7.3). Defines the basic activities that an organization (that is, acquirer, supplier, developer, operator, maintainer, or the manager of another process) performs for establishing, measuring, controlling, and improving its life cycle process.
- 4) *Training process* (subclause 7.4). Defines the activities for providing adequately trained personnel.

4.1.2 Tailoring process. Annex A, which is normative, defines the basic activities needed to perform tailoring of this International Standard. Annex B contains a brief guidance on tailoring the requirements of this International Standard; it lists the key factors upon which tailoring decisions may be made.

4.1.3 Relationship between the processes and organizations

This International Standard contains various processes that are applied throughout the life cycle of software by various organizations depending on their needs and goals. For understandability, annex C presents the relationships between the life cycle processes and related parties.

5 Primary life cycle processes

This clause defines the following primary life cycle processes:

- 1) Acquisition process;
- 2) Supply process;
- 3) Development process;
- 4) Operation process;
- 5) Maintenance process.

The activities and tasks in a primary process are the responsibility of the organization initiating and performing that process. This organization ensures that the process is in existence and functional.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 12207:1995](https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995)

<https://standards.iteh.ai/catalog/standards/sist/ce835b8a-a9bf-423a-8250-f3f9047d3ffb/iso-iec-12207-1995>

5.1 Acquisition process

The Acquisition Process contains the activities and tasks of the acquirer. The process begins with the definition of the need to acquire a system, software product or software service. The process continues with the preparation and issue of a request for proposal, selection of a supplier, and management of the acquisition process through to the acceptance of the system, software product or software service.

The individual organization having the need may be called the owner. The owner may contract any or all of the acquisition activities to an agent who will in turn conduct these activities according to the Acquisition Process. The acquirer in this subclause may be the owner or the agent.

The acquirer manages the Acquisition Process at the project level following the Management Process (7.1), which is instantiated in this process; establishes an infrastructure under the process following the Infrastructure Process (7.2); tailors the process for the project following the Tailoring Process (annex A); and manages the process at the organizational level following the Improvement Process (7.3) and the Training Process (7.4).

List of activities: This process consists of the following activities:

- 1) Initiation;
- 2) Request-for-Proposal [-tender] preparation;
- 3) Contract preparation and update;
- 4) Supplier monitoring;
- 5) Acceptance and completion.

5.1.1 Initiation. This activity consists of the following tasks:

5.1.1.1 The acquirer begins the acquisition process by describing a concept or a need to acquire, develop, or enhance a system, software product or software service.

5.1.1.2 The acquirer will 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.

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

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

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

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

- a) Purchase an off-the-shelf software product that satisfies the requirements.
- b) Develop the software product or obtain the software service internally.
- c) Develop the software product or obtain the software service through contract.
- d) A combination of a, b, and c above.
- e) Enhance an existing software product or service.

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

- a) The requirements for the software product are satisfied.
- b) The documentation is available.
- c) Proprietary, usage, ownership, warranty and licensing rights are satisfied.
- d) Future support for the software product is planned.

5.1.1.8 The acquirer should prepare, document and execute an acquisition plan. The plan should contain the following:

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

5.1.1.9 The acquirer should define and document the acceptance strategy and conditions (criteria).

5.1.2 Request-for-proposal [-tender] preparation. This activity consists of the following tasks:

5.1.2.1 The acquirer should document the acquisition requirements (e.g., request for proposal), the content of which depends upon the acquisition option selected in 5.1.1.6. The acquisition documentation should include, as appropriate:

- a) System requirements;
- b) Scope statement;
- c) Instructions for bidders;
- d) List of software products;
- e) Terms and conditions;
- f) Control of subcontracts;
- g) Technical constraints (e.g., target environment).

5.1.2.2 The acquirer should determine which processes, activities, and tasks of this International Standard are appropriate for the project and should tailor them accordingly. Especially, the acquirer should specify the applicable supporting processes (clause 6) and their performing organizations, including responsibilities (if other than supplier), so that the suppliers may, in their proposals, define the approach to each of the specified supporting processes. The acquirer will define the scope of those tasks that reference the contract.

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

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

5.1.3 Contract preparation and update. This activity consists of the following tasks:

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

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

5.1.3.3 The acquirer may involve other parties, including potential suppliers, before contract award, in tailoring this International Standard for the project. However, the acquirer will make the final decision on the tailoring. The acquirer will include or reference the tailored International Standard in the contract.

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