



**Systems and software engineering —
Software life cycle processes**

STANDARDS
Australia



Currently in preview, click buy full version

AS ISO/IEC/IEEE 12207:2019

This Australian Standard® was prepared by IT-015, Software and Systems Engineering. It was approved on behalf of the Council of Standards Australia on 25 September 2019.

This Standard was published on 11 November 2019.

The following are represented on Committee IT-015:

- Australian Computer Society
- Australian Digital Health Agency
- Australian Society for Technical Communication
- Department of Defence (Australian Government)
- Engineers Australia
- Griffith University
- IT Service Management Forum Australia
- NSW Business Chamber
- University of New South Wales
- University of Southern Queensland
- University of Technology Sydney

This Standard was issued in draft form for comment as DR AS ISO/IEC/IEEE 12207:2019.

Keeping Standards up-to-date

Ensure you have the latest versions of our publications and keep up-to-date about Amendments, Rulings, Withdrawals, and new projects by visiting:

www.standards.org.au

ISBN 978 1 76072 603 4



Systems and software engineering — Software life cycle processes

Originates as AS/NZS ISO/IEC 12207:1997.
Previous edition 2013.
Revised and redesignated as AS ISO/IEC/IEEE 12207:2019.

COPYRIGHT

© ISO/IEC/IEEE 2019 — All rights reserved
© Standards Australia Limited 2019

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher, unless otherwise permitted under the Copyright Act 1968 (Cth).

Preface

This Standard was prepared by the Australian members of Joint Standards Australia/Standards New Zealand Committee IT-015, Systems and Software Engineering, to supersede AS/NZS ISO/IEC 12207:2013.

After consultation with stakeholders in both countries, Standards Australia and Standards New Zealand decided to develop this Standard as an Australian Standard rather than an Australian/New Zealand Standard.

The objective of this Standard is to establish 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 applicable during the acquisition, supply, development, operation, maintenance or disposal of software systems, products, and services. These life cycle processes are accomplished through the involvement of stakeholders, with the ultimate goal of achieving customer satisfaction.

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

This document also provides a set of processes to facilitate communication among acquirers, suppliers and other stakeholders in the life cycle of a software system. It also provides processes that can be employed for defining, controlling, and improving software life cycle processes within an organization or a project.

The processes, activities, and tasks of this document can also be applied during the acquisition or supply of a system that contains software, either alone or in conjunction with AS/NZS ISO/IEC/IEEE 15288:2015, *Systems and software engineering — System life cycle processes*.

This Standard is identical with, and has been reproduced from, ISO/IEC/IEEE 12207:2017, *Systems and software engineering — Software life cycle processes*.

As this document has been reproduced from an International Standard, a full point substitutes for a comma when referring to a decimal marker.

Australian or Australian/New Zealand Standards that are identical adoptions of international normative references may be used interchangeably. Refer to the online catalogue for information on specific Standards.

The terms “normative” and “informative” are used in Standards to define the application of the appendices or annexes to which they apply. A “normative” appendix or annex is an integral part of a Standard, whereas an “informative” appendix or annex is only for information and guidance.

Contents

| | |
|--|-----------|
| Preface | ii |
| Foreword | v |
| Introduction | vii |
| 1 Scope | 1 |
| 1.1 Overview | 1 |
| 1.2 Purpose | 1 |
| 1.3 Field of application | 1 |
| 1.4 Limitations | 2 |
| 2 Normative references | 2 |
| 3 Terms, definitions, and abbreviated terms | 2 |
| 3.1 Terms and definitions | 2 |
| 3.2 Abbreviated terms | 12 |
| 4 Conformance | 12 |
| 4.1 Intended usage | 12 |
| 4.2 Full conformance | 13 |
| 4.2.1 Full conformance to outcomes | 13 |
| 4.2.2 Full conformance to tasks | 13 |
| 4.3 Tailored conformance | 14 |
| 5 Key concepts and application | 14 |
| 5.1 Introduction | 14 |
| 5.2 Software system concepts | 14 |
| 5.2.1 Software systems | 14 |
| 5.2.2 Software system structure | 15 |
| 5.2.3 Enabling systems | 16 |
| 5.2.4 Life cycle processes for the software system | 17 |
| 5.3 Organization and project concepts | 18 |
| 5.3.1 Organizations | 18 |
| 5.3.2 Organization and project-level adoption | 19 |
| 5.4 Life cycle concepts | 19 |
| 5.4.1 Software life cycle stages | 19 |
| 5.4.2 Life cycle model for the software system | 19 |
| 5.5 Process concepts | 21 |
| 5.5.1 Criteria for processes | 21 |
| 5.5.2 Description of processes | 21 |
| 5.5.3 General characteristics of processes | 21 |
| 5.5.4 Tailoring | 21 |
| 5.6 Process groups | 22 |
| 5.6.1 Introduction | 22 |
| 5.6.2 Agreement processes | 23 |
| 5.6.3 Organizational project-enabling processes | 24 |
| 5.6.4 Technical Management processes | 24 |
| 5.6.5 Technical processes | 24 |
| 5.7 Process application | 25 |
| 5.8 Process reference model | 26 |
| 6 Software life cycle processes | 26 |
| 6.1 Agreement processes | 26 |
| 6.1.1 Acquisition process | 26 |
| 6.1.2 Supply process | 29 |
| 6.2 Organizational Project-Enabling processes | 31 |
| 6.2.1 Life cycle model management process | 32 |
| 6.2.2 Infrastructure Management process | 33 |

| | | |
|---------------------|---|------------|
| 6.2.3 | Portfolio Management process..... | 34 |
| 6.2.4 | Human Resource Management process..... | 36 |
| 6.2.5 | Quality Management process..... | 37 |
| 6.2.6 | Knowledge Management process..... | 39 |
| 6.3 | Technical Management processes..... | 41 |
| 6.3.1 | Project Planning process..... | 41 |
| 6.3.2 | Project assessment and control process..... | 44 |
| 6.3.3 | Decision Management process..... | 46 |
| 6.3.4 | Risk Management process..... | 48 |
| 6.3.5 | Configuration Management process..... | 50 |
| 6.3.6 | Information Management process..... | 55 |
| 6.3.7 | Measurement process..... | 56 |
| 6.3.8 | Quality Assurance process..... | 58 |
| 6.4 | Technical processes..... | 60 |
| 6.4.1 | Business or Mission Analysis process..... | 61 |
| 6.4.2 | Stakeholder Needs and Requirements Definition process..... | 64 |
| 6.4.3 | System/Software requirements definition process..... | 68 |
| 6.4.4 | Architecture Definition process..... | 73 |
| 6.4.5 | Design Definition process..... | 78 |
| 6.4.6 | System Analysis process..... | 81 |
| 6.4.7 | Implementation process..... | 83 |
| 6.4.8 | Integration process..... | 87 |
| 6.4.9 | Verification process..... | 90 |
| 6.4.10 | Transition process..... | 94 |
| 6.4.11 | Validation process..... | 98 |
| 6.4.12 | Operation process..... | 102 |
| 6.4.13 | Maintenance process..... | 105 |
| 6.4.14 | Disposal process..... | 110 |
| Annex A | (normative) Tailoring process..... | 113 |
| Annex B | (informative) Examples of process information items..... | 115 |
| Annex C | (informative) Process Reference Model for assessment purposes..... | 120 |
| Annex D | (informative) Process integration and process constructs..... | 122 |
| Annex E | (informative) Process views..... | 124 |
| Annex F | (informative) Software system architecture modelling..... | 134 |
| Annex G | (informative) Application of software life cycle processes to a system of systems..... | 137 |
| Annex H | (informative) Application of Agile..... | 141 |
| Annex I | (informative) Process Mapping from ISO/IEC/IEEE 12207:2008..... | 143 |
| Bibliography | | 159 |

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.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO, IEC, and IEEE shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Systems and software engineering*, in cooperation with the IEEE Computer Society Systems and Software Engineering Standards Committee, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

This first edition of ISO/IEC/IEEE 12207 cancels and replaces ISO/IEC 12207:2008 (second edition), which has been technically revised.

Changes in this revision of ISO/IEC/IEEE 12207 were developed in conjunction with a corresponding revision of ISO/IEC/IEEE 15288:2015, *Systems and software engineering — System life cycle processes*. The purpose of these revisions is to accomplish the harmonization of the structures and contents of the two documents, while supporting the requirements of the engineering and assessment communities.

This document was developed with the following goals:

- provide a common terminology between the revision of ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207;
- where applicable, provide common process names and process structure between the revision of ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207; and
- enable the user community to evolve towards fully harmonized standards, while allowing backward compatibility.

This revision is intended to achieve a fully harmonized view of the system and software life cycle processes.

Currently in preview, click buy full version

Introduction

The complexity of software systems has increased to an unprecedented level. This has led to new opportunities, but also to increased challenges for the organizations that create and utilize systems. These challenges exist throughout the life cycle of a system and at all levels of architectural detail. This document provides a common process framework for describing the life cycle of systems created by humans, adopting a Software Engineering approach. Software Engineering is an interdisciplinary approach and means to enable the realization of successful software systems. It focuses on defining stakeholder needs and required functionality early in the development cycle, documenting requirements, and performing design synthesis and system validation while considering the complete problem. It integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation and maintenance. It considers both the business and the technical needs of all stakeholders with the goal of providing a quality product that meets the needs of users and other applicable stakeholders. This life cycle spans the conception of ideas through to the retirement of a system. It provides the processes for acquiring and supplying systems. It helps to improve communication and cooperation among the parties that create, utilize and manage modern software systems in order that they can work in an integrated, coherent fashion. In addition, this framework provides for the assessment and improvement of the life cycle processes.

The processes in this document form a comprehensive set from which an organization can construct software life cycle models appropriate to its products and services. An organization, depending on its purpose, can select and apply an appropriate subset to fulfill that purpose.

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

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

Australian Standard[®]

Systems and software engineering — Software life cycle processes

1 Scope

1.1 Overview

This document 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 applicable during the acquisition, supply, development, operation, maintenance or disposal of software systems, products, and services. These life cycle processes are accomplished through the involvement of stakeholders, with the ultimate goal of achieving customer satisfaction.

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

This document also provides processes that can be employed for defining, controlling, and improving software life cycle processes within an organization or a project.

The processes, activities, and tasks of this document can also be applied during the acquisition of a system that contains software, either alone or in conjunction with ISO/IEC/IEEE 15288:2015, *Systems and software engineering—System life cycle processes*.

In the context of this document and ISO/IEC/IEEE 15288, there is a continuum of human-made systems from those that use little or no software to those in which software is the primary interest. It is rare to encounter a complex system without software, and all software systems require physical system components (hardware) to operate, either as part of the software system-of-interest or as an enabling system or infrastructure. Thus, the choice of whether to apply this document for the software life cycle processes, or ISO/IEC/IEEE 15288:2015, *Systems and software engineering—System life cycle processes*, depends on the system-of-interest. Processes in both documents have the same process purpose and process outcomes, but differ in activities and tasks to perform software engineering or systems engineering, respectively.

1.2 Purpose

The purpose of this document is to provide a defined set of processes to facilitate communication among acquirers, suppliers, and other stakeholders in the life cycle of a software system.

This document is written for acquirers, suppliers, developers, integrators, operators, maintainers, managers, quality assurance managers, and users of software systems, products, and services. It can be used by a single organization in a self-imposed mode or in a multi-party situation. Parties can be from the same organization or from different organizations and the situation can range from an informal agreement to a formal contract.

The processes in this document can be used as a basis for establishing business environments, e.g., methods, procedures, techniques, tools and trained personnel. [Annex A](#) provides normative direction regarding the tailoring of these software life cycle processes.

1.3 Field of application

This document applies to the full life cycle of software systems, products, and services, including conception, development, production, utilization, support and retirement, and to their acquisition and supply, whether performed internally or externally to an organization. The life cycle processes of this document can be applied concurrently, iteratively and recursively to a software system and incrementally to its elements.