
**Systems and software engineering —
Software Engineering Environment
Services**

*Ingénierie du logiciel et des systèmes — Services d'environnement en
ingénierie du logiciel*

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 15940:2013](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013)

[https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-
1116de295f88/iso-iec-15940-2013](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013)

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 15940:2013](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013)

<https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013>



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2013

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword	vi
Introduction.....	vii
1 Scope	1
2 Terms and definitions	1
3 Abbreviated terms	3
4 Reference Model for SEE services	3
4.1 Categories of SEE services	3
4.2 Structure of service description	3
4.3 Reference model.....	3
5 Software engineering services	5
5.1 Overview.....	5
5.2 Software requirements engineering service.....	6
5.3 Software reverse engineering service.....	6
5.4 Software re-engineering service.....	7
5.5 Software prototyping service.....	7
5.6 Software modelling service.....	7
5.7 Software simulation service.....	8
5.8 Software design service.....	8
5.9 Software component based software generation service.....	8
5.10 Software source code generation service.....	9
5.11 Software compilation service.....	9
5.12 Software debugging service.....	9
5.13 Software static/dynamic analysis service.....	10
5.14 Software testing service	10
5.15 Software verification service.....	11
5.16 Software integration service	11
5.17 Software domain engineering service (Software reuse)	11
5.18 Software reuse asset management service (Software reuse)	12
5.19 Software reuse program management service (Software reuse).....	12
6 Systems engineering services	13
6.1 Overview.....	13
6.2 System solution orientation service.....	13
6.3 System Operational scenarios service	14
6.4 System modelling service	14
6.5 System architectural design service	15
6.6 System Requirements engineering service.....	15
6.7 System Re-engineering service	15
6.8 System Simulation service.....	16
6.9 System Integration service.....	16
6.10 System testing service.....	16
6.11 System test synthesis & report service	17
6.12 System work product verification service	17
7 System engineering techniques services.....	18
7.1 Overview.....	18
7.2 Value analysis service	18
7.3 Trade-off analysis.....	18
7.4 Effectiveness analysis	19
7.5 Technology maturity analysis.....	19

8	Technical management services.....	20
8.1	Overview	20
8.2	Configuration management service.....	20
8.3	Change management service	20
8.4	SEE repository management service	21
8.5	Reuse management service	21
8.6	Measurement and analysis service.....	21
8.7	Quality assurance service	22
8.8	Audit service	22
8.9	Traceability service	22
8.10	Documentation service	23
8.11	Review service support.....	23
9	Project management services	24
9.1	Overview	24
9.2	Project strategy service	24
9.3	Project planning service	24
9.4	Project estimation service	25
9.5	Project risk management service.....	25
9.6	Project monitoring and scheduling service.....	25
9.7	Project evaluation service	26
9.8	Decision management service	26
9.9	Information management service	27
10	Process management services	27
10.1	Overview	27
10.2	Process definition service	28
10.3	Process library service	28
10.4	Process initiation service	28
10.5	Process usage service	29
10.6	Process monitoring service.....	29
10.7	Process improvement support service	29
10.8	Process documentation service.....	30
11	SEE support services	30
11.1	Overview	30
11.2	SEE common support service.....	30
11.3	SEE publishing service	31
11.4	SEE cooperative work support service	31
11.5	SEE user communication support service	31
11.6	SEE administration service	32
11.7	SEE policy enforcement service	32
11.8	SEE data/information mining service	32
11.9	SEE data retrieve/storage service.....	33
11.10	SEE data/information exchange service	33
11.11	SEE enabling support service	33
12	SEE infrastructure services	34
12.1	Overview	34
12.2	SEE infrastructure management service	34
12.3	SEE information sharing service	35
12.4	SEE repository service.....	35
12.5	SEE operating system service	35
Annex A	(informative) Exemplary automated support for the SEE services	37
Annex B	(informative) Services mapped on to ISO/IEC 12207 activities.....	46
Annex C	(informative) Services mapped on to ISO/IEC 15288 activities.....	52
Annex D	(informative) Exemplary categories relationship for the SEE Services	57
Annex E	(informative) Application of this International Standard	58

ITC STANDARD PREVIEW
 (standards.itech.ai)
 ISO/IEC 15940:2013
<https://standards.itech.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013>

Annex F (informative) Traceability between ISO 15940:2006 and “ISO-15940 for Systems and Software engineering”	59
Bibliography.....	63

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 15940:2013](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013)

<https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013>

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.

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 15940 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

This second edition cancels and replaces the first edition (ISO/IEC 15940:2006), which has been technically revised.

ISO/IEC 15940:2013
<https://standards.iteh.ai/catalog/standards/sist/cfl71808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013>

Introduction

Software engineering environments, or “SEEs” refer to a collection of services, partially or fully automated by software tools, that are used to support the execution of human activities in systems and software engineering.

These activities are usually carried out within a software or system development/maintenance project, and cover such areas as the specification, development, re-engineering or maintenance of systems.

ISO/IEC 12207 describes in a comprehensive manner all of the processes, activities and tasks performed during the software life cycle.

The term “Software Engineering Environment” may cover several situations; from the mere juxtaposition of a few tools running on the same operating system, to the fully integrated environment, able to handle, monitor, and even control all the data, processes, and activities in the systems and software engineering life cycle. A SEE provides support to human activities through a series of services that describe the capabilities of the environment. The software process supported by a SEE becomes an assisted or automated software process. This International Standard describes SEE services and relates them to ISO/IEC 12207:2008 in a manner applicable to a range of organizations. In defining a life cycle process for an organization, the user needs to find the appropriate level of automation provided by a software engineering environment. This may result in establishing a new SEE or improving an existing one.

Through the automation of activities, either partially or fully, the SEE provides benefits to an organization through reduced cost (higher productivity), improved management and from the higher product quality that can result. For example, the automation of repetitive activities such as the execution of test cases provides not only productivity gains, but can also help to ensure completeness and consistency in the testing activities

This International Standard defines the SEE services conceptually in a reference model that can be adapted to any SEEs to automate one or more software and system engineering activities.

For a user interested in a specific process, this International Standard describes the relationship between given systems and software engineering processes, the software engineering services, and the corresponding exemplary software engineering tools.

The suite of SEE services described supports the process definitions in ISO/IEC 12207. The purpose is to define a set of SEE Services that are compatible with ISO/IEC 12207:2008, and that can be used either as a general reference, or to define an automated software and system process.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 15940:2013](#)

<https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-1116de295f88/iso-iec-15940-2013>

Systems and software engineering — Software Engineering Environment Services

1 Scope

This International Standard provides a description of SEE services that supports all of the software and system life cycle processes defined in ISO/IEC 12207.

The services are intended as a complete set and can be used in any systems and software engineering development or support organization where there is a need to select one or more SEE services. Such an organization may or may not have systems and software projects that use the ISO/IEC 12207 process framework.

A reference model for SEE Services is provided within this International Standard. This reference model has been produced starting from References [8] and [9]. This International Standard was produced using material originally published by the Software engineering Institute (Carnegie Mellon University, USA), NIST and ECMA, which finally resulted in a joint effort from ECMA and NIST indicating a broad consensus at the time of publication. In addition to this background process, structure from ISO/IEC 12207:2008 has been used as a baseline.

SEE Services for System Engineering and Software Reuse have been added to the revision of this International Standard. The list of changes between the previous version (ISO/IEC 15940:2006) and this version is given in Annex F.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

2.1

life cycle model

framework containing the processes, activities and tasks involved in the development operation and maintenance of a software and system product, spanning the life of the system from the definition of its requirements to the termination of its use

[ISO/IEC 12207:2008]

2.2

CASE

computer-aided software engineering

use of computers to aid in the software engineering process

[ISO/IEC/IEEE 24765:2010]

2.3

CASE tool

software product that can assist software and system engineers by providing automated support for software and system engineering life-cycle activities as defined in ISO/IEC 12207:2008

[ISO/IEC 14102:2008]

**2.4
organization**

group of people and facilities with an arrangement of responsibilities, authorities and relationships

[ISO 9000:2005]

**2.5
work product**

any artefact produced by a process

NOTE This may include files, documents, part of the product, services, processes, specifications, and invoices.

[ISO/IEC/IEEE 24765:2010]

**2.6
systems engineering**

interdisciplinary approach governing the total technical and managerial effort required to transform a set of customer needs, expectations, and constraints into a solution and to support that solution throughout its life

NOTE This may include the definition of technical performance measures; the integration of engineering specialties toward the establishment of an architecture; and the definition of supporting lifecycle processes that balance cost, performance and schedule objectives

[ISO/IEC/IEEE 24765:2010]

**2.7
software engineering environment**

SEE

provides automated system context services and software specific services for the engineering of software systems and related domains (e.g., project management, process management, etc.)

NOTE It includes the platform, system software, utilities, and CASE tools installed.

**2.8
SEE Service**

consists in one or more service operations to support life cycle activities for the SEE

NOTE A SEE Service supplier provides a SEE Service for a SEE Service acquirer.

**2.9
automated or assisted systems or software process**

systems or software process that is performed either fully or partially supported by CASE tools

**2.10
actor**

organization or CASE tool that supplies and/or acquires SEE Services

**2.11
operation**

action needed to perform an Activity

NOTE One or more operations are necessary to execute an Activity. An operation may consist of other operations.

**2.12
SEE Service acquirer**

actor that acquires a SEE Service

**2.13
SEE Service supplier**

actor that supplies a SEE Service

3 Abbreviated terms

CASE – Computer Aided Software Engineering

SEE – Software Engineering Environment.

4 Reference Model for SEE services

4.1 Categories of SEE services

This International Standard provides a reference model for SEE services. As a reference model, this International Standard uses a set of conceptual descriptions to describe each service used in a software engineering environment. The “conceptual description” indicates that the description is from a reference viewpoint, and does not deal with any specific implementation. The description is therefore general and does not assume any specific application domain, life cycle model, or tool in a project. In this way, this International Standard can be applied to any defined organizational environment.

An actual environment is one that is built from a reference model containing conceptual descriptions. Therefore, an actual description of a specific environment would reflect a particular activity with its tools and standards. The services described in this International Standard are grouped into eight categories that reflect broad functional activities within a typical systems and software engineering organization. The eight categories are:

- Software engineering services (e.g., Software Modelling);
- Systems engineering services (e.g., System Modelling);
- Systems engineering techniques services (e.g. Value analysis);
- Technical management services (e.g., Reuse, Configuration management);
- Project management services (e.g., Estimation, Project monitoring);
- Process management services (e.g., Process Monitoring, Process improvement);
- SEE Support services (e.g., Publishing, Policy enforcement);
- SEE infrastructure services (e.g., Repository, Communication, Operating System services).

4.2 Structure of service description

Each service is defined under two headings:

- Service Concept, to provide a description of the service in terms that are not related to a specific implementation;
- Service Operations, to list those operations that may be included in a service. These lists of operational capabilities represent, in most cases, primary services only and are not intended to be complete.

Exemplary automated supports for each SEE Services are listed in Annex A, it includes lists of corresponding service operations to help readers understand SEE.

4.3 Reference model

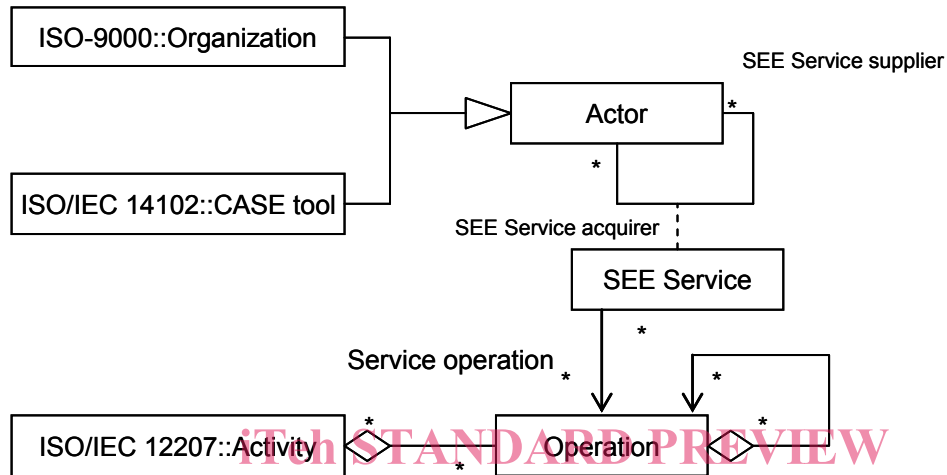
SEE services can be identified within a Reference model. This section presents those concepts that are part of this reference model (see Fig.1 SEE Reference Model described in UML). The reference model is made of the following concepts:

- Software and system engineering Environment (model itself);
- SEE Service;
- SEE Service Operation;

ISO/IEC 15940:2013(E)

- CASE Tool;
- Actor;
- Activity;
- Organization.

While engineering software and systems and in related domains (e.g. project management), a life cycle Activity is achieved by one or more Operations. SEE Service operations satisfy target life cycle activities. Actor provides and/or consumes SEE Service.



NOTE Actor itself can be acquirer and/or supplier of a SEE service that is provided by an association of Actors. SEE Service can not exist alone without Actor.

ISO/IEC 15940:2013

[https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-116d795f88/iso-iec-15940-2013)

[116d795f88/iso-iec-15940-2013](https://standards.iteh.ai/catalog/standards/sist/cf171808-4391-4cef-bc4d-116d795f88/iso-iec-15940-2013)

NOTE brief usages of UML notation are described here for readers benefit

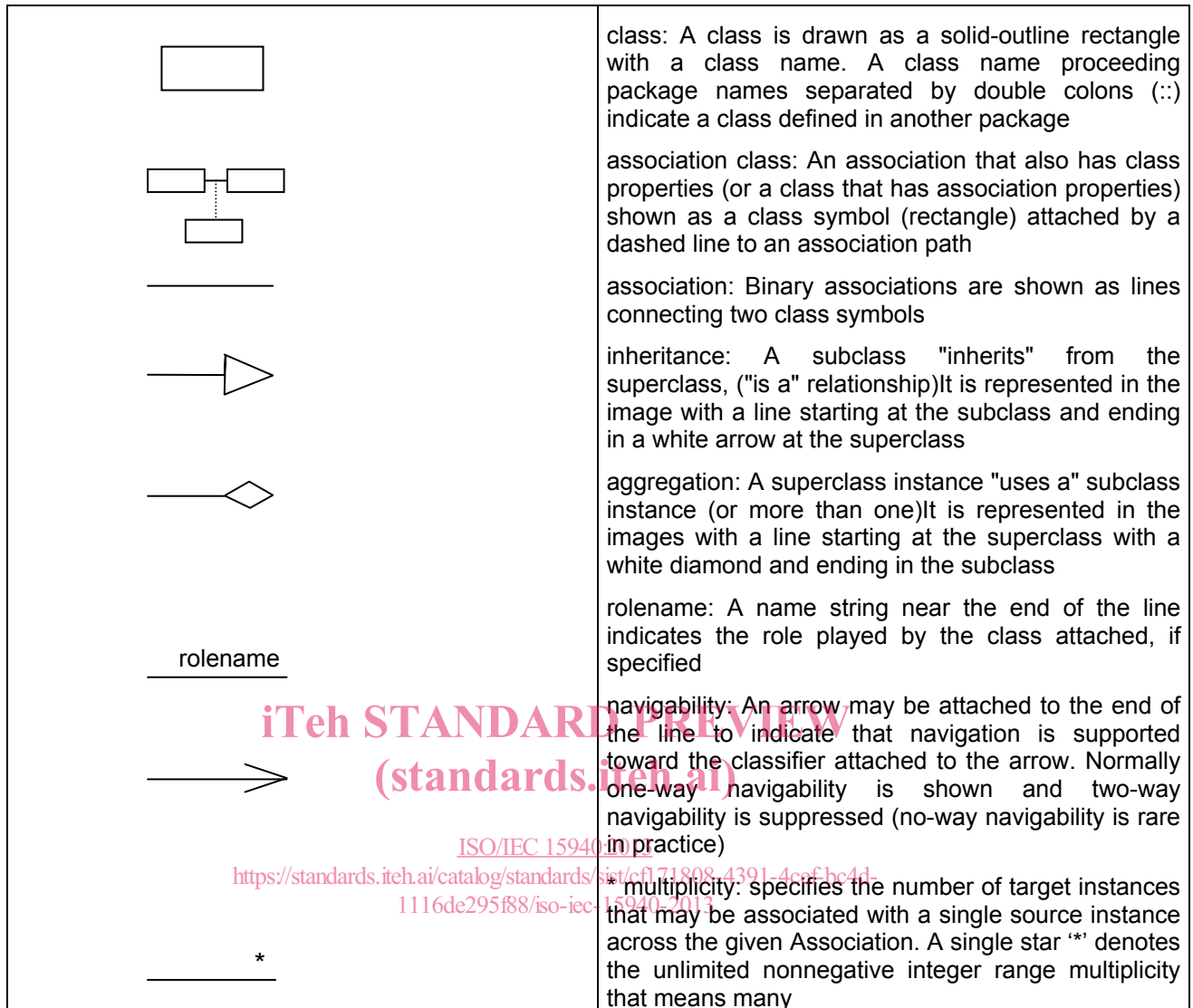


Figure 1 — SEE Reference Model described in UML

5 Software engineering services

5.1 Overview

The services in this section support activities related to software life cycle specific services. The following services are defined and grouped in this section:

- * Software requirements engineering services;
- * Software reverse engineering service;
- * Software re-engineering service;
- * Software prototyping service;
- * Software design service;
- * Software modelling service;
- * Software simulation service;
- * Software component based software generation service;

- * Software source code generation service;
- * Software compilation service;
- * Software debugging service;
- * Software static/dynamic analysis service;
- * Software testing services;
- * Software verification service;
- * Software integration services;
- * Software domain engineering service (Software reuse);
- * Software reuse asset management service (Software reuse);
- * Software reuse program management service (Software reuse).

5.2 Software requirements engineering service

5.2.1 Service concept

This service provides the ability to capture, represent, analyse, validate, and refine those system requirements that fulfil operational needs and are allocated to software components.

5.2.2 Service operations

This service provides the ability to:

- Elicit and capture software and business requirements;
- Structure the software requirements;
- Create, modify, browse, and present software requirements;
- Group and prioritise software requirements;
- Check consistency of software requirements;
- Allocate software requirements for each software component;
- Conduct impact analysis for the addition, subtraction, or modification in a requirement against the project, value, resources, and timeline;
- Validate and baseline the document specs based on stakeholders and developers.

5.3 Software reverse engineering service

5.3.1 Service concept

This service provides the ability to capture design information from source or object code, and produce structure charts, call graphs, and other design documentation to provide new functionality or support a new environment.

5.3.2 Service operations

This service provides the ability to:

- Generate design from source code;
- Generate source program from object code;

- Generating design metrics (complexity metrics etc) (so that they can be considered for decision for reengineering or rewrite);
- Identifying unused code so that it can be removed if necessary.

5.4 Software re-engineering service

5.4.1 Service concept

This service provides the ability to take a new or a modified set of software requirements and the existing design as input and produce a new or modified design.

5.4.2 Service operations

This service provides the ability to:

- Revise or restructure existing design, components and/or code;
- Perform impact analysis of new design on existing software components;
- Translate from one notation or language into another;
- Check that the new set of requirements is consistent with the existing system;
- Determine the impact of the altered design on the existing set of components;
- Creating reusable modules from existing code;
- Creating wrappers for existing modules.

5.5 Software prototyping service

5.5.1 Service concept

This service provides the ability to enable the production of a software system that reproduces the user interface and emulates the functionality and behaviour of the final system to be built.

5.5.2 Service operations

This service provides the ability to:

- Build a prototype from requirements;
- Develop reusable or throw away prototype;
- Document the findings from prototype;
- Develop sample prototype of reports required;
- Invoke software modelling service if necessary and available;
- Produce a user interface from requirements;
- Execute a prototype : Also include capture user feedback on prototype;
- Conduct simulations if necessary and available.

5.6 Software modelling service

5.6.1 Service concept

This service provides the ability to model requirements and/or design in order to determine the effectiveness of alternative designs with respect to such attributes as user interface characteristics or execution flow.