
**Software and systems engineering —
Tools and methods for product line
requirements engineering**

*Ingénierie du logiciel et des systèmes — Outils et méthodes pour
l'ingénierie d'exigences pour gammes de produits*

This document is a preview generated by EVS



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2012

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing 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	v
Introduction.....	vi
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Reference model for product line requirements engineering.....	3
5 Product Line Scoping	7
5.1 Product scoping	7
5.1.1 Structure information to be used for scoping	8
5.1.2 Identify products	8
5.1.3 Analyze common and variable features	9
5.1.4 Define a product portfolio.....	9
5.2 Domain scoping.....	10
5.2.1 Identify functional domains.....	10
5.2.2 Map features to functional domains.....	10
5.2.3 Define domain scope	11
5.3 Asset scoping	11
5.3.1 Gather historical data from existing single products	12
5.3.2 Estimate additional effort required to adapt potential assets	12
5.3.3 Estimate expected development effort for new products in the product portfolio definition.....	13
5.3.4 Estimate economic benefits from reusing proposed assets	13
5.3.5 Derive asset proposals from economic evaluation results	13
6 Domain Requirements Engineering	14
6.1 Domain requirements elicitation.....	14
6.1.1 Draw a context diagram	15
6.1.2 Gather domain information	15
6.1.3 Identify initial domain requirements.....	16
6.1.4 Review the elicited initial domain requirements	16
6.2 Domain requirements analysis	17
6.2.1 Classify and balance initial domain requirements.....	17
6.2.2 Analyze commonalities and variabilities	18
6.2.3 Model domain requirements	18
6.2.4 Create prototypes and analyze feasibility.....	19
6.2.5 Develop conceptual test cases and scenarios for acceptance testing	19
6.2.6 Review the analyzed domain requirements.....	19
6.3 Domain requirements specification	20
6.3.1 Identify sources of domain requirements.....	20
6.3.2 Establish traceability.....	21
6.3.3 Document domain requirements	21
6.3.4 Review the domain requirements specification.....	22
6.4 Domain requirements verification and validation.....	22
6.4.1 Verify domain requirements	23
6.4.2 Validate domain requirements	23
6.4.3 Validate conceptual test cases and scenarios for acceptance testing	23
6.4.4 Baseline domain requirements	24
6.4.5 Initiate change management process	24
6.5 Domain requirements management	24
6.5.1 Manage domain requirements change	25
6.5.2 Manage traceability	26

6.5.3	Manage versions of domain requirements	26
6.5.4	Record and report status	26
6.5.5	Manage process improvement	27
6.5.6	Manage feedback	27
7	Variability Management in Requirements Engineering	27
7.1	Variability in textual requirements	28
7.1.1	Define requirements variability in textual requirements	28
7.1.2	Document requirements variability in textual requirements	28
7.2	Variability in requirements models	29
7.2.1	Define requirements variability in model	29
7.2.2	Document requirements variability in requirements model	30
7.3	Traceability between requirements variability and variability model	30
7.3.1	Define explicit links between requirements variability and variability model	30
8	Asset Management in Requirements Engineering	31
8.1	Domain requirements artifacts as domain assets	31
8.1.1	Identify domain requirements artifacts managed as domain assets	32
8.1.2	Define configuration and annotation	32
8.2	Application requirements artifacts as application assets	32
8.2.1	Identify application requirements artifacts managed as application assets	33
8.2.2	Define configuration and annotation for application requirements assets	33
9	Application Requirements Engineering	34
9.1	Application requirements elicitation	34
9.1.1	Draw a context diagram for an application	35
9.1.2	Identify the requirements gaps between domain and application requirements	35
9.1.3	Bind the best matching variants	36
9.1.4	Select domain assets	36
9.1.5	Review the elicited application requirements	36
9.2	Application requirements analysis	37
9.2.1	Classify and balance application specific initial requirements	38
9.2.2	Analyze commonalities and variabilities	38
9.2.3	Model application specific requirements	38
9.2.4	Create prototypes and analyze feasibility	39
9.2.5	Develop conceptual test cases and scenarios for acceptance testing	39
9.2.6	Review the analyzed application requirements	40
9.3	Application requirements specification	40
9.3.1	Identify sources of application specific requirements	41
9.3.2	Establish traceabilities for application specific requirements	41
9.3.3	Document application specific requirements	41
9.3.4	Document the rationale for variability decision	42
9.3.5	Review the application requirements specification	42
9.4	Application requirements verification and validation	42
9.4.1	Verify application specific requirements	43
9.4.2	Validate application specific requirements	43
9.4.3	Validate conceptual test cases and scenarios for acceptance testing	43
9.4.4	Baseline application specific requirements	44
9.4.5	Initiate application change management process	44
9.5	Application requirements management	44
9.5.1	Manage application specific requirements change	45
9.5.2	Manage application specific traceability	46
9.5.3	Manage versions of application specific requirements artifacts	46
9.5.4	Record and report status of application requirements management	46
9.5.5	Manage application specific process improvement	47
Annex A (informative)	Comparison of requirements engineering tasks between single product and product line	48
Annex B (informative)	A Construct for Process, Method, Tool, and Aspect	51
Bibliography	52

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electro-technical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member 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 shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 26551 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

Introduction

The major purpose of this International Standard is to deal with the capabilities of tools and methods of software and systems product line (SSPL) requirements engineering. This International Standard defines how the tools and methods can support for the software and systems product line-specific requirements engineering processes.

Decision for the initial boundaries of domain is conducted in advance by defining a product line scope before initiating domain requirements engineering processes. Domain requirements engineering will be carried out in a comprehensive manner because common and variable requirements and captured variabilities have consequential impacts on member products in a product line. The outcomes of domain requirements engineering processes are transferred into the requirements of a family of products at the application requirements engineering processes. Therefore requirements engineering tools and methods should consider both engineering processes namely domain requirements engineering and application requirements engineering.

Product line requirements engineering can be differentiated from a single product requirement engineering because of the following aspects:

- There are two core processes in requirements engineering: domain requirements engineering and application requirements engineering. The major aims of the domain requirements engineering processes are to analyze commonality and variability for a family of products, and to prepare necessary variability information for variability modelling. The major aims of the application requirements engineering processes are to define application specific requirements and bind variability defined in domain requirements engineering processes.
- It is essential to analyse the costs and benefits estimation of a product line and thereby an organization can make a go/no-go decision. Moreover, the costs and benefits estimation plays a pivotal role as an indicator for assessing the effectiveness and efficiency of a product line.

This International Standard can be used in the following modes:

- By the users of this International Standard – to benefit people who develop, operate, and manage requirements engineering for software and systems product lines.
- By a product line organization – to provide guidance in the evaluation and selection for tools and methods for product line requirements engineering.
- By providers of tools and methods – to provide guidance in implementing or developing tools and methods by providing a comprehensive set of the capabilities of tools and methods for product line requirements engineering.

ISO/IEC 26550 (ISO/IEC 26550, Software and systems engineering — Reference model for product line engineering and management) addresses both engineering and management processes and covers the key characteristics of product line development. ISO/IEC 26550 provides an overview of the consecutive international standards (i.e., ISO/IEC 26551 through ISO/IEC 26556) as well as the structure of the model:

- Processes and capabilities of methods and tools for product line scoping, domain requirements engineering, and application requirements engineering are provided as ISO/IEC 26551, *Software and systems engineering — Tools and methods for product line requirements engineering*.
- Processes and capabilities of methods and tools for domain design and application design are provided as ISO/IEC 26552, *Software and systems engineering — Tools and methods for product line architecture design*.

- Processes and capabilities of methods and tools for domain realization and application realization are provided as ISO/IEC 26553, *Software and systems engineering — Tools and methods for product line realization*.
- Processes and capabilities of methods and tools for domain verification and validation and application verification and validation are provided as ISO/IEC 26554, *Software and systems engineering — Tools and methods for product line verification and validation*.
- Processes and capabilities of methods and tools for technical management are provided as ISO/IEC 26555, *Software and systems engineering — Tools and methods for product line technical management*.
- Processes and capabilities of methods and tools for organizational management are provided as ISO/IEC 26556, *Software and systems engineering — Tools and methods for product line organizational management*.

Software and systems engineering — Tools and methods for product line requirements engineering

1 Scope

This International Standard deals with the tools and methods of requirements engineering for software and systems product line. The scope of this International Standard is as follows:

- provide the terms and definitions specific to requirements engineering for software and systems product lines.
- define process groups and their processes performed during product line requirements engineering. Those processes are described in terms of purpose, inputs, tasks, and outcomes.
- define method capabilities to support the defined tasks of each process.
- define tool capabilities to automate/semi-automate tasks or defined method capabilities.

This International Standard does not concern processes and capabilities of requirements tools and methods for a single system but rather deals with those for a family of products.

NOTE This International Standard is not suitable for handling physical artifacts. In the Systems arena, the word "Product" must be understood as System-level artefacts, such as requirement documents, architectural data, validation plans, Behavioral Models, etc. In any case, the word "Product" must not be understood as physical items such as electronic boards, mechanical parts or qualified human operators. In the case of the Software components of a Systems, this International Standard can apply twice: once to handle the System-Level Product Line and a second time to handle the Software Part Product Line, if any. The Product Line processes are recursive within the different levels of Products.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 26550, *Software and systems engineering — Reference model for product line engineering and management*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 26550 and the following apply following terms and definitions apply.

3.1

application assets in requirements

application specific artifacts produced during application requirements engineering such as application requirements specifications and application requirements models

3.2

application requirements elicitation

identifies stakeholders relevant to an application, elicits application specific requirements, and binds the appropriate variants