INTERNATIONAL STANDARD

ISO/IEC 26557

First edition
2016-12-15

Software and systems engineering — Methods and tools for variability mechanisms in software and systems product line

Ingénierie du logiciel et des systèmes — Méthodes et outils pour les mécanismes de variabilité dans les chaînes de production de logiciels et de systèmes

Reference number
ISO/IEC 26557:2016(E)

© ISO/IEC 2016



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2016, Published in Switzerland

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
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword ... **vi**

Introduction ... **vii**

1 Scope ... **1**

2 Normative references ... **1**

3 Terms and definitions ... **1**

4 Variability mechanisms for software and systems product line (SSPL) ... **3**
4.1 Overview ... 3
4.2 Reference model for variability mechanisms in product line ... 6

5 Variability mechanism management ... **8**
5.1 Variability mechanism planning ... 9
5.1.1 Purpose of variability mechanism planning ... 9
5.1.2 Estimate adequate resources needed for variability mechanism operationalization ... 9
5.1.3 Assign responsibility for variability mechanism operationalization ... 10
5.1.4 Defining quality assurance measures for variability mechanism operationalization ... 10
5.2 Variability mechanism enabling ... 11
5.2.1 Purpose of variability mechanism enabling ... 11
5.2.2 Enable variability mechanism pool ... 12
5.2.3 Provide guidance for variability mechanism operationalization ... 12
5.2.4 Enable measurement infrastructure for quantifying variability mechanism operationalization ... 12
5.2.5 Procure resources needed to perform variability mechanism operationalization ... 13
5.3 Variability mechanism tracking ... 13
5.3.1 Purpose of variability mechanism tracking ... 13
5.3.2 Review the plan versus actual of variability mechanism operationalization ... 14
5.3.3 Assess issues in variability mechanism operationalization ... 14
5.3.4 Make corrective actions for variability mechanism operationalization ... 15

6 Variability mechanism operationalization ... **15**
6.1 Variability mechanism operationalization for requirements ... 16
6.1.1 Purpose of variability mechanism operationalization for requirements ... 16
6.1.2 Categorize requirements variability ... 16
6.1.3 Assess requirements level variability mechanism ... 17
6.1.4 Specify requirements level variability mechanism ... 17
6.1.5 Prepare bindings at requirements level ... 18
6.1.6 Verify requirements level variability mechanism ... 18
6.2 Variability mechanism operationalization for design ... 19
6.2.1 Purpose of variability mechanisms in domain design ... 19
6.2.2 Make architectural decisions on binding times ... 20
6.2.3 Assess variability mechanisms depending on the binding time ... 20
6.2.4 Define guides and rules on variability mechanisms in architectural texture ... 20
6.2.5 Specify architectural variability mechanisms ... 21
6.2.6 Prepare bindings at architecture level ... 21
6.2.7 Verify architectural variability mechanisms ... 22
6.3 Variability mechanism operationalization for realization ... 22
6.3.1 Purpose of variability mechanisms in domain realization ... 22
6.3.2 Examine architectural decisions and architectural texture on realization ... 23
6.3.3 Assess detailed design level variability mechanisms ... 24
6.3.4 Specify detailed design level variability mechanisms ... 24
6.3.5 Define post-detailed design guides on variability mechanisms ... 25
6.3.6 Verify detailed design level variability mechanisms ... 25

© ISO/IEC 2016 – All rights reserved

6.3.7 Assess implementation level variability mechanisms ... 26
6.3.8 Specify implementation level variability mechanisms ... 26
6.3.9 Enable implementation level configurability ... 26
6.3.10 Prepare bindings at realization time ... 27
6.3.11 Verify implementation level variability mechanisms ... 27
6.4 Variability mechanism operationalization at compile time ... 28
6.4.1 Purpose of variability mechanism operationalization at compile time ... 28
6.4.2 Examine architectural decisions and architectural texture on compile time ... 28
6.4.3 Assess compile time variability mechanisms ... 29
6.4.4 Specify compile time variability mechanisms ... 29
6.4.5 Enable compile time configurability ... 30
6.4.6 Prepare bindings at compile time ... 30
6.4.7 Verify compile time variability mechanisms ... 30
6.5 Variability mechanism operationalization at post-compile time ... 31
6.5.1 Purpose of variability mechanism operationalization at post-compile time ... 31
6.5.2 Examine architectural decisions and architectural texture affecting post-compile time ... 32
6.5.3 Assess post-compile time variability mechanisms ... 32
6.5.4 Specify link time variability mechanisms ... 32
6.5.5 Specify load time variability mechanisms ... 33
6.5.6 Specify deployment time variability mechanisms ... 33
6.5.7 Enable post-compile time configurability ... 33
6.5.8 Prepare bindings at post-compile time ... 34
6.5.9 Verify post-compile time variability mechanism ... 34
6.6 Variability mechanism operationalization at run time ... 35
6.6.1 Purpose of variability mechanism operationalization at run time ... 35
6.6.2 Examine architectural decisions and architectural texture affecting run time reconfiguration ... 35
6.6.3 Assess run time variability mechanism ... 36
6.6.4 Enable run time configurability ... 36
6.6.5 Prepare bindings at run time ... 36
6.6.6 Verify run time variability mechanism ... 37
6.7 Variability mechanism operationalization for test artefacts ... 37
6.7.1 Purpose of variability mechanism operationalization for test artefacts ... 37
6.7.2 Examine test strategy on variability mechanisms ... 38
6.7.3 Assess the decisions on variability mechanisms of requirements, architecture and realization ... 38
6.7.4 Specify variability mechanisms in each test level ... 39
6.7.5 Enable reusability in testing ... 39
6.7.6 Prepare bindings at test stage ... 39
6.7.7 Verify variability mechanism operationalization for test artefacts ... 40

7 Variability mechanism support ... 40
7.1 Relating variability mechanism to variability model ... 41
7.1.1 Purpose of relating variability mechanism to variability model ... 41
7.1.2 Relate variability mechanism to variability model ... 41
7.1.3 Add annotation to relationship ... 42
7.2 Quality assurance for variability mechanism ... 42
7.2.1 Purpose of quality assurance for variability mechanism ... 42
7.2.2 Objectively evaluate variability mechanism activities ... 43
7.2.3 Objectively evaluate variability mechanism work products ... 43
7.2.4 Communicate and resolve non-compliance issues ... 44
7.2.5 Establish records of variability mechanism quality assurance activities ... 44
7.3 Binding time decision support ... 44
7.3.1 Purpose of binding time decision support ... 44
7.3.2 Determine the value of decision variables on a decision table ... 45
7.3.3 Specify decisions on binding time ... 45
7.3.4 Verify the decision table ... 45
7.4 Application configuration support ... 46

 © ISO/IEC 2016 – All rights reserved

7.4.1 Purpose of application configuration support 46
7.4.2 Support realizing configurability 46
7.4.3 Apply decision rules for configuration 47
7.4.4 Improve configurability 47
Annex A (informative) **Variability mechanisms in software development activities** **48**
Annex B (informative) **Binding time decision from variability types** **49**
Bibliography **50**

© ISO/IEC 2016 – All rights reserved

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.

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 ISO documents 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 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.

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

© ISO/IEC 2016 – All rights reserved

Introduction

Software and systems product line (SSPL) engineering and management creates, exploits and manages a common platform to develop a family of products (e.g. software products, systems architectures) at lower cost, reduced time to market and with better quality. As a result, it has gained increasing global attention since 1990s.

Variability, which differentiates a member product from other products within a product line, plays an important role in SSPL. Variability mechanism means ways to implement variability; it realizes variability in the product line artefacts. Variability mechanisms differ in accordance with the binding time of variability, and variability of a product line is introduced from product line scoping through product line testing and its binding can occur at any stages of product line development. Thus, variability mechanism should be systematically managed for the right operation in domain engineering and for the right binding in application engineering. Furthermore, variability mechanisms should support easy variability management and traceability management. Accordingly, this document provides processes with their supporting methods and tools capabilities for variability mechanism operationalization and for managerial supports for the right use of variability mechanisms at domain engineering stages and the right bindings at application engineering stages.

This document can be used in the following modes:

— by the users of this document: to benefit people who want to adopt SSPL for producing their products by guiding variability mechanism operationalization, variability mechanism management and variability mechanism supports;

— by a product line organization: to provide guidance in the evaluation and selection for methods and tools for the tasks of providing variability mechanism operationalization, variability mechanism management and variability mechanism supports;

— 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 methods and tools for supporting variability mechanism operationalization, variability mechanism management and variability mechanism supports.

The ISO/IEC 26550 family of standards addresses both engineering and management processes and capabilities of methods and tools in terms of the key characteristics of product line development. This document provides processes and capabilities of methods and tools for variability mechanisms in product lines. Other ISO/IEC 26550 family of standards are as follows.

ISO/IEC 26550, ISO/IEC 26551 and ISO/IEC 26555 are published. ISO/IEC 26558 and ISO/IEC 26559 are under preparation. ISO/IEC 26552, ISO/IEC 26553, ISO/IEC 26554, ISO/IEC 26556, ISO/IEC 26560, ISO/IEC 26561, ISO/IEC 26562 and ISO/IEC 26563 are planned.

— Processes and capabilities of methods and tools for domain requirements engineering and application requirements engineering are provided in ISO/IEC 26551.

— Processes and capabilities of methods and tools for domain design and application design are provided in ISO/IEC 26552 (planned).

— Processes and capabilities of methods and tools for domain realization and application realization are provided in ISO/IEC 26553 (planned).

— Processes and capabilities of methods and tools for domain testing and application testing are provided in ISO/IEC 26554 (planned).

— Processes and capabilities of methods and tools for technical management are provided in ISO/IEC 26555.

— Processes and capabilities of methods and tools for organizational management are provided in ISO/IEC 26556 (planned).

© ISO/IEC 2016 – All rights reserved

— Processes and capabilities of methods and tools for variability modelling are provided in ISO/IEC 26558.

— Processes and capabilities of methods and tools for variability traceability are provided in ISO/IEC 26559.

— Processes and capabilities of methods and tools for product management are provided in ISO/IEC 26560 (planned).

— Processes and capabilities of methods and tools for technical probe are provided in ISO/IEC 26561 (planned).

— Processes and capabilities of methods and tools for transition management are provided in ISO/IEC 26562 (planned).

— Processes and capabilities of methods and tools for configuration management of asset are provided in ISO/IEC 26563 (planned).

— Others (ISO/IEC 26564 to ISO/IEC 26599): planned.

© ISO/IEC 2016 – All rights reserved

Software and systems engineering — Methods and tools for variability mechanisms in software and systems product line

1 Scope

This document, within the context of tools and methods of variability mechanisms for software and system product lines**:**

— provides the terms and definitions related to variability mechanisms for software and systems product lines;

— defines processes and their subprocesses for operating variability mechanisms at each product line life cycle stages and those for providing managerial supports. Those processes are described in terms of purpose, inputs, tasks and outcomes;

— defines method capabilities to support the defined tasks of each process;

— defines tool capabilities to automate/semi-automate tasks or defined method capabilities.

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

2 Normative references

There are no normative references in this document.

3 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp/

3.1
application configuration
derivation for a member product specific executables from domain assets in *realization* (3.10)

Note 1 to entry: The specific configuration of an application is the *binding* (3.3) results for the *variation points* (3.19) with the selected *variants* (3.17).

3.2
aspect
special consideration within product line engineering process groups and tasks to which we can associate specialized methods and tools

3.3
binding
task for making a *decision* (3.7) on relevant *variants* (3.17) using domain *variability model* (3.16) and *decision tables* (3.8)

© ISO/IEC 2016 – All rights reserved

3.4
binding time
moment of variability resolution

Note 1 to entry: The choice of binding time is independent from variability modelling. It is the consequence of *decisions* (3.7) made from requirements through *run time* (3.11). Demands for flexibility and the support of tools allow late binding times or even the use of variable binding times.

3.5
binding time decision
selection for *variability* (3.13) defined in platforms in accordance with the functional distinction between variability in time and variability in space

3.6
configurability
degree of how well a *variability mechanism* (3.14) supports the configuration of a member product

3.7
decision
types of statements in which a choice between two or more possible outcomes controls which set of actions will result

3.8
decision table
table that specifies decision variables

Note 1 to entry: It also includes rules, constraints and relevancy among variables.

3.9
post-compile time
collective name for link time and load time that are right after the compilation of components

3.10
realization
stage for detailed design and construction

3.11
run time
stage that a member product is executed

Note 1 to entry: Components can be developed, compiled, linked and loaded separately. Only at run time are they combined into a working system.

3.12
texture
architectural texture
collection of common development rules and constraints for realizing the applications of a product line

3.13
variability
characteristics that may differ among members of a product line

Note 1 to entry: The differences between members may be captured from multiple viewpoints such as functionality, quality attributes, environments in which the members are used, users, constraints and internal mechanisms that realize functionality and quality attributes.

Note 2 to entry: It is important to distinguish between the concepts of system and software variability and product line variability. Any system partially or fully composed of software can be considered to possess software variability because software systems are inherently malleable, extendable or configurable for specific use contexts. Product line variability is concerned with the variability that is explicitly defined by product management. This document is primarily concerned with product line variability.

© ISO/IEC 2016 – All rights reserved

EXAMPLE 1 In the case of a home automation system, in accordance with business strategy, the use of a LAN as an alternative to the European Installation Bus (EIB) in a home automation system might be a competitive advantage for the company since it allows the use of low-cost components.

EXAMPLE 2 Annex B provides variability examples in accordance with variability types.

3.14
variability mechanism
variability representation/implementation technique for the product line variability

Note 1 to entry: It deals with variabilities based on the *binding time* (3.4) at the specific life cycle stage.

3.15
variability mechanism operationalization
VMO
adequate provision or *binding* (3.3) of *variability mechanisms* (3.14) at each specific domain or application engineering life cycle stage

3.16
variability model
explicit definition for product line variability

Note 1 to entry: It introduces *variation points* (3.19), types of variation for the variation points, *variants* (3.17) offered by the variation points, variability dependencies and variability constraints. Variability models may be orthogonal to or integrated in other models such as requirements or design models. There are two types of variability models: application variability models and domain variability models.

3.17
variant
option or an alternative that may be used to realize particular *variation points* (3.19)

Note 1 to entry: One or more variants should correspond to each variation point to represent V_VP relationship. Selection and *binding* (3.3) of variants for a specific product determine the characteristics of the particular *variability* (3.13) for the product.

3.18
variant selection
decision making for a choice of a *variant* (3.17) in a *variation point* (3.19)

Note 1 to entry: *binding* (3.3), variability resolution.

3.19
variation point
representation corresponding to particular variable characteristics of products, domain assets and application assets in the context of a product line

Note 1 to entry: Variation points show which of the product line element varies. Each variation point can have multiple V_VP relationships.

4 Variability mechanisms for software and systems product line (SSPL)

4.1 Overview

Variability mechanism means a method for implementing the variability of a product line and it incorporates variability into the product line development artefacts. Because variability is introduced at the whole product line life cycle stages and so binding does, variability mechanisms for implementing variability in accordance with its binding time should be provided. It is necessary to classify variability mechanisms by binding times so the users of this document can choose proper variability mechanisms in accordance with the binding time decisions. This subclause provides a set of variability mechanisms used at given binding times.

© ISO/IEC 2016 – All rights reserved