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

*Ingénierie des systèmes et du logiciel — Méthodes et outils pour
modéliser la traçabilité dans les gammes de produits des logiciels et
systèmes*





COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2017, 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 | v |
| Introduction | vi |
| 1 Scope | 1 |
| 2 Normative references | 1 |
| 3 Terms and definitions | 1 |
| 4 Abbreviated terms | 2 |
| 5 Reference model for variability traceability in software and systems product line | 2 |
| 5.1 Overview | 2 |
| 5.2 Reference model for variability traceability in software and systems product line | 4 |
| 6 Variability tracing management | 7 |
| 6.1 General | 7 |
| 6.2 Variability tracing planning | 7 |
| 6.2.1 Purpose of variability tracing planning | 7 |
| 6.2.2 Design variability tracing strategy | 8 |
| 6.2.3 Define quality assurance measures for variability tracing | 8 |
| 6.2.4 Assign responsibility for variability tracing | 9 |
| 6.2.5 Record variability tracing plan | 9 |
| 6.3 Variability tracing enabling | 10 |
| 6.3.1 Purpose of variability tracing enabling | 10 |
| 6.3.2 Provide guidance for variability tracing | 10 |
| 6.3.3 Mobilize roles and responsibilities for variability tracing | 11 |
| 6.3.4 Enable variability tracing management | 11 |
| 6.3.5 Enable variability tracing operations | 11 |
| 6.3.6 Enable quality assurance measurement for variability tracing | 12 |
| 6.4 Variability tracing managing | 12 |
| 6.4.1 Purpose of variability tracing managing | 12 |
| 6.4.2 Review the plan versus actual result of variability tracing | 13 |
| 6.4.3 Control issues on domain/application variability tracing | 13 |
| 6.4.4 Control issues on variability traceability management | 13 |
| 6.4.5 Control issues on variability tracing support | 14 |
| 6.4.6 Support corrective actions for variability tracing | 14 |
| 6.4.7 Make improvement actions for variability tracing | 15 |
| 7 Variability tracing | 15 |
| 7.1 General | 15 |
| 7.2 Traceability between domain variability model and domain artefacts | 15 |
| 7.2.1 Purpose of traceability between domain variability model and domain artefacts | 15 |
| 7.2.2 Identify relations between variation point and domain artefact | 16 |
| 7.2.3 Identify relations between variant and domain artefact | 16 |
| 7.2.4 Define the trace links between domain variability model and domain artefacts | 17 |
| 7.3 Traceability between application variability model and application artefacts | 17 |
| 7.3.1 Purpose of traceability between application variability model and application artefacts | 17 |
| 7.3.2 Identify relations between variation point and application artefact | 18 |
| 7.3.3 Identify relations between variant and application artefact | 18 |
| 7.3.4 Define the trace links between application variability model and application artefacts | 19 |
| 7.4 Traceability between domain and application variability model | 19 |
| 7.4.1 Purpose of traceability between domain and application variability model | 19 |
| 7.4.2 Identify relations between domain and application variability model | 20 |
| 7.4.3 Define trace links in accordance with binding results | 20 |
| 7.5 Traceability among variability related artefacts within a stage | 20 |

| | | |
|----------|---|-----------|
| 7.5.1 | Purpose of traceability between variability related artefacts within a stage..... | 20 |
| 7.5.2 | Identify relations among domain artefacts within the same stage..... | 21 |
| 7.5.3 | Define trace links among domain artefacts within the same stage..... | 21 |
| 7.5.4 | Instantiate trace links in line with binding results..... | 22 |
| 7.6 | Traceability between variability related artefacts in adjacent stages..... | 22 |
| 7.6.1 | Purpose of traceability between variability related artefacts in different adjacent stages..... | 22 |
| 7.6.2 | Identify relations between domain artefacts in adjacent stages..... | 23 |
| 7.6.3 | Define trace links between domain artefacts in adjacent stages..... | 23 |
| 7.6.4 | Instantiate trace links in line with binding results..... | 24 |
| 8 | Variability tracing support..... | 24 |
| 8.1 | General..... | 24 |
| 8.2 | Relating tracing to variability model..... | 24 |
| 8.2.1 | Purpose of relating tracing to variability model..... | 24 |
| 8.2.2 | Identify variability tracing conflicts..... | 25 |
| 8.2.3 | Harmonize variability traceability with product line traceability..... | 25 |
| 8.3 | Quality assurance for variability tracing..... | 26 |
| 8.3.1 | Purpose of quality assurance for variability tracing..... | 26 |
| 8.3.2 | Objectively evaluate variability tracing activities..... | 26 |
| 8.3.3 | Objectively evaluate variability tracing work products..... | 27 |
| 8.3.4 | Communicate and resolve noncompliance issues..... | 27 |
| 8.3.5 | Establish records of tracing quality assurance activities..... | 28 |
| 8.4 | Variability tracing algorithm..... | 28 |
| 8.4.1 | Purpose of variability tracing algorithm..... | 28 |
| 8.4.2 | Identify algorithms for variability tracing..... | 29 |
| 8.4.3 | Evaluate defined variability tracing algorithms..... | 29 |
| 8.4.4 | Measure the performance of algorithms..... | 30 |
| 8.4.5 | Optimize traceability algorithms..... | 30 |
| | Annex A (informative) Variability traceability dimension..... | 31 |
| | Annex B (informative) Types of variability traceability relations..... | 32 |
| | Annex C (informative) Variability traceability attributes..... | 34 |
| | Bibliography..... | 35 |

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 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 and IEC 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 voluntary nature of standards, 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, *Software and systems engineering*.

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 the 1990s.

Variability, which differentiates a member product from other products within a product line, plays an important role in SSPL; and hundreds of variabilities are introduced throughout the whole SSPL domain engineering stages. Those variabilities are defined, refined, newly added as domain engineering stages go forward. Variabilities thus are modelled carefully so as to manage and control them in a systematic way. This document deals with methods and tools capability for supporting variability modelling using consistent notations and for managing and/or utilizing variability models in domain and application engineering lifecycle processes.

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 how to model variabilities among member products;
- by a product line organization: to provide guidance in the evaluation and selection for methods and tools for variability modelling;
- by providers of tools and methods: to provide guidance in implementing or developing methods and tools by providing a comprehensive set of methods and tools capabilities for supporting variability modelling.

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 modelling in product lines. Other ISO/IEC 26550 family of standards are as follows:

- 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;
- processes and capabilities of methods and tools for domain realization and application realization are provided in ISO/IEC 26553;
- processes and capabilities of methods and tools for domain testing and application testing are provided in ISO/IEC 26554;
- 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;
- processes and capabilities of methods and tools for variability mechanisms are provided in ISO/IEC 26557;
- 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;
- processes and capabilities of methods and tools for technical probe are provided in ISO/IEC 26561;

- processes and capabilities of methods and tools for transition management are provided in ISO/IEC 26562;
- processes and capabilities of methods and tools for configuration management of asset are provided in ISO/IEC 26563;
- others (ISO/IEC 26564 to ISO/IEC 26599): to be developed.

ISO/IEC 26550, ISO/IEC 26551 and ISO/IEC 26555 are published. ISO/IEC 26557 and ISO/IEC 26559 are to be published. ISO/IEC 26552, ISO/IEC 26553, ISO/IEC 26554, ISO/IEC 26556, ISO/IEC 26560, ISO/IEC 26561, ISO/IEC 26562, ISO/IEC 26563 are planned International Standards.

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

1 Scope

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

- provides the terms and definitions specific to variability traceability for software and systems product lines;
- defines process groups and their processes for establishing and managing variability traceability at product line lifecycle processes. 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 <http://www.iso.org/obp>

3.1

aspect

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

3.2

instantiated trace link

trace link (3.6) derived by applying binding in a member product

3.3

texture

architectural texture

collection of common development rules and constraints for realising the applications of a product line

3.4

traceability

discernible association among two or more logical entities, such as requirements, system elements, verifications, or tasks

3.5
traceability with different abstraction level
level of detail for established *traceability* (3.4)

EXAMPLE The finer level of detail, the more general level.

3.6
trace link
association between two trace artefacts or between an element of variability model and a development artefact

3.7
trace link semantics
purpose or meaning of the *trace link* (3.6), specified in the trace link types

EXAMPLE ‘Implements’, ‘refines’, ‘requires’ and ‘excludes’.

3.8
variability traceability
traceability (3.4) among variability definition, variability implementation in domain engineering and application engineering, and thereafter tracing impacts due to variability changes

3.9
variant
instance or value of a *variation point* (3.10)

3.10
variation point
indication of product differentiation based on particular variable characteristics of products, domain assets, and application assets in the context of a product line

4 Abbreviated terms

| | |
|------|-----------------------------------|
| CRUD | create, read, update, delete |
| OVM | orthogonal variability model |
| SSPL | software and systems product line |

5 Reference model for variability traceability in software and systems product line

5.1 Overview

Variability is a key differentiator between single-system engineering and management and product line engineering and management. Product line engineering and management has to take explicitly into account the variations within and between multiple products. The product line variabilities are introduced and defined during product management, domain engineering and application engineering processes defined in ISO/IEC 26550. Explicitly defined variability is represented or implemented by different development artifacts, and it should be traced for the proper management. Variability traceability enables tracing variabilities defined in variability models from their introduction stage to binding stage. Variability tracing should be able to trace variability with all of relevant development artefacts of domain engineering and application engineering processes of ISO/IEC 26550.

SSPL traceability provides a set of linkages of artefacts in a product line. Compared to the traceability of single product development, SSPL traceability is more complex because SSPL has two separate but intimately related development processes, i.e. domain engineering and application engineering. SSPL traceability includes trace links between artefacts within the same stages, trace links between

artefacts produced in different stages, trace links between domain and application artefacts, and trace links among variability-related artefacts. This document deals with trace links among variability-related artefacts, i.e. variability traceability.

Variability traceability enables the establishment and maintenance of trace links among variabilities described in the various forms of domain assets, application assets, and variability models. Variability traceability is important for the consistent maintenance and evolution of variabilities implemented/described in those various forms. Trace links can be simple relationships between different variability-related artefacts or they can be comprehensive and sophisticated trace information including trace link semantics. Because having too much (comprehensive) or too little (simple) traceability is costly or insufficient, an economic analysis for the abstraction level of traceability should be conducted in order to make decision on the appropriate level of details for traceability establishment and management.

The variability of a product line is spread all over development artefacts, so it is difficult to trace variability, for example, between the sources of variability and the corresponding artefacts. An orthogonal variability model (OVM), a separated model defining the variability of a product line, provides a cross-sectional view of the variability across all development artefacts. It relates the elements of a variability defined to different development artefacts such as feature models, requirements artefacts, architecture, detailed designs, codes, test artefacts, and after compile time artefacts (e.g. makefile). Thus, the variability of different development artefacts can be traced through an OVM. [Figure 1](#) illustrates the concept of OVM-based variability traceability in domain engineering.

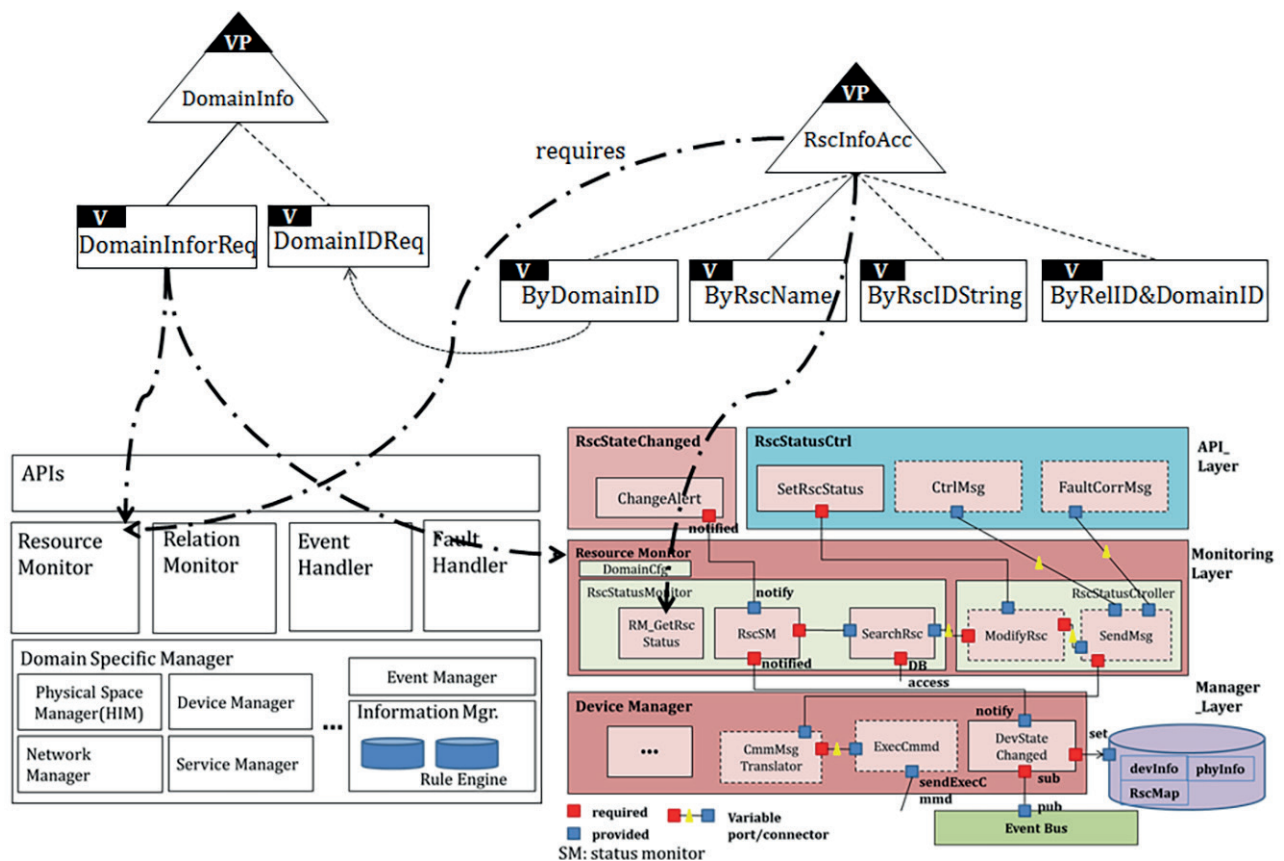


Figure 1 — OVM-based variability tracing in domain engineering

Variability relating to several domain development artefacts can be traced through the OVM. As shown in [Figure 1](#), a variation point or variant can relate to different development artefacts of different stages. They can relate to a use case, corresponding sub-systems, and corresponding components. Therefore, relations among artefacts corresponding to the same variability either located in the same stage or different stages can be traced through the links from the elements of the OVM. Because the trace links from the OVM to development artefacts include links from a variation point to its representing artefacts

and links from the variants to their realizing artefacts, impacts on the corresponding artefacts due to the variability changes or evolution can be completely and easily analysed.

The OVM, namely the domain variability model, relates with the application variability model in accordance with binding decisions and the corresponding trace links are also instantiated along with binding. Member product specific variability is defined through the application variability model and it relates to the corresponding application artefacts. From introduction time through binding time, variability traceability is achieved through the instantiated trace links, trace links between the OVM and application variability model, and member product specifically established trace links. [Figure 2](#) illustrates OVM based variability tracing in a product line.

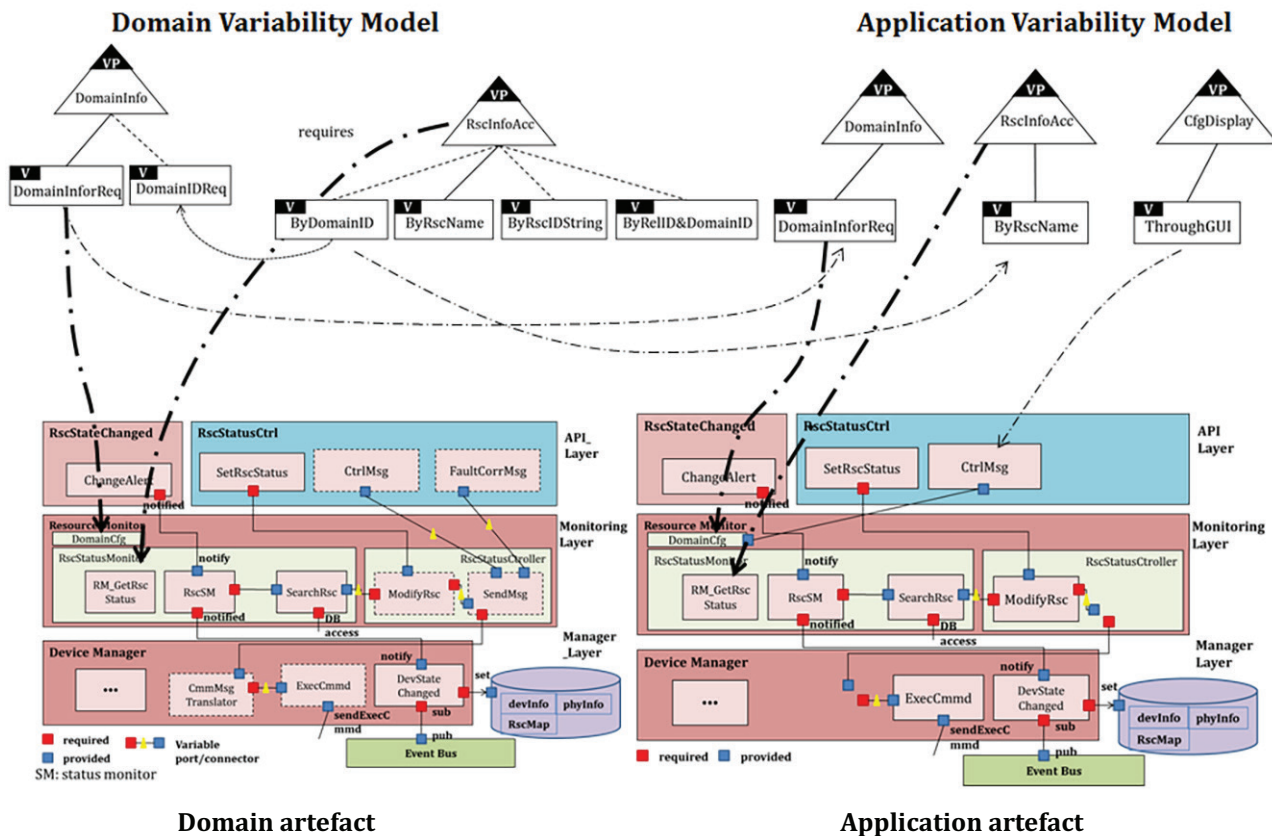


Figure 2 — OVM based variability tracing within a product line

The thick broken line means instantiated trace links in accordance with binding decisions in application engineering. There exists a newly added variability (e.g. CfgDisplay in [Figure 2](#)) and its trace link with application artefacts is defined.

5.2 Reference model for variability traceability in software and systems product line

The reference model specifies the structure of supporting processes and subprocesses for variability traceability in product line. As shown in [Figure 3](#), variability traceability in product line can be structured into three processes: variability tracing management, variability tracing and variability tracing support. In the rest of this document, tasks, methods and tools are described in terms of processes and subprocesses defined in the reference model.

Each process is divided into subprocesses and each subprocess is described in terms of the following attributes:

- the title of the subprocess;
- the purpose of the subprocess;

- the inputs to produce the outcomes;
- the tasks to achieve the outcomes;
- the outcomes of the subprocess;
- the capabilities of tools and methods are a list of the required support of tools and methods for performing the tasks properly.

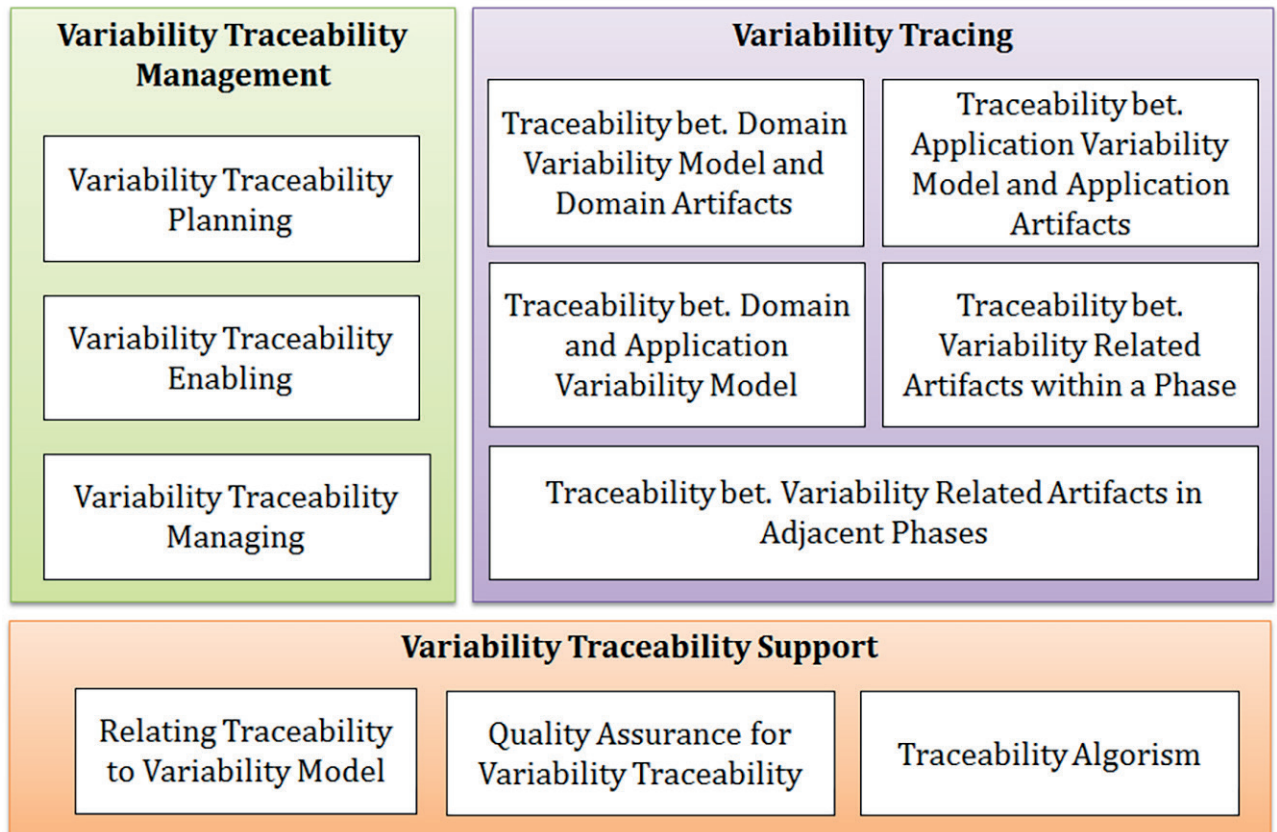


Figure 3 — Variability traceability in SSPL

The variability tracing management process supports for establishing plans for establishing and maintaining variability traceability with the different levels of detail, for providing enabling environments for realizing the planned variability traceability, and for managing plans versus actual variability traceability status. The variability tracing management process provides required principles and foundations such as the level of abstraction of traceability, traceability notation, and traceability algorithm for establishing and maintaining variability traceability. Variability tracing management shall do the following:

- variability tracing planning;
- variability tracing enabling;
- variability tracing managing.

The variability tracing process supports establishing and maintaining variability traceability in domain engineering and application engineering. Variability tracing shall do the following:

- traceability between domain variability model and domain artefacts;
- traceability between application variability model and application artefacts;
- traceability between domain and application variability model;

- traceability between variability related artefacts within a stage;
- traceability between variability related artefacts in different adjacent stages.

The variability tracing support process provides required support for establishing and maintaining the required variability traceability including traceability algorithm, validation and verification supports for established variability traceability, and traceability with variability models. Variability tracing support shall do the following:

- relating traceability to variability model;
- quality assurance for variability tracing;
- traceability algorithm.

The identification and analysis of the key differentiators between single-system engineering and management and product line engineering and management can help organizations to understand the product line and to formulate a strategy for successful implementation of product line engineering and management. The key aspects have been defined in ISO/IEC 26550 and [Table 1](#) shows the categories of the key aspects.

Table 1 — Key aspects for identifying product line-specific variability traceability tasks

| Category | Aspects |
|------------------------|--|
| Reuse management | Application engineering, domain assets, domain engineering, product management, platform, reusability |
| Variability management | Binding, variability |
| Complexity management | Collaboration, configuration, enabling technology support, reference architecture, texture, traceability |
| Quality management | Measurement and tracking, cross functional verification and validation |

The following are the descriptions for each aspect concerning variability traceability for product lines. The variability tracing relevant processes and tasks shall be identified on the basis of these aspects. The concerns specific to variability traceability for product lines will enable an organization to understand the variability tracing relevant processes, subprocesses, tasks, methods and tools' capabilities.

- Application engineering: Variabilities-bound and newly-added application specific variabilities during application engineering should be traced. Application variability model plays a key role for maintaining those traceabilities.
- Binding: Binding results (i.e. variants decision) in a member product should be properly traced so as to manage them with variability changes and evolutions.
- Collaboration: Variability traceability enables collaboration among domain and application engineers.
- Configuration: Bound variability of a configuration should be traced and its version should be properly managed.
- Domain asset: Variabilities implemented through domain assets should be traced. Variability model is used to trace them.
- Domain engineering: Variability defined and implemented during domain engineering should be traced. Trace links for a variability implemented in different domain assets and trace links for a variability implemented throughout the different domain engineering stages should be established and maintained. Domain variability model can be used to trace a variability implemented throughout the different domain engineering stages.
- Enabling technology support: Technologies for establishing and maintaining trace links for variabilities should be supported.

- Measurement and tracking: The effectiveness and efficiency of traceability established and maintained should be properly measured and tracked so as to improve variability tracing processes.
- Platform: Variability consisting parts of a platform should be traced.
- Product management: Variability traceability should evolve in accordance with the evolution and changes of product lines. Variability traceability should be established and maintained towards ease of supporting the evolution and changes of product lines.
- Reference architecture: Reference architecture has a key role in a product line, so variability traceability for variability-relevant architectural design decisions, architectural elements, and textures in the reference architecture should be carefully established and managed.
- Reusability: Variability traceability enables the reusability of variability among member products under the change and evolution control of variability.
- Texture: Texture related to rules and constraints for variability should be properly traced by variability traceability.
- Traceability: Traceability among variability expressed/implemented in requirements, architecture, realization, and variability models should be established and managed.
- Cross functional validation and verification: Validation and verification for established and maintained variability traceability are performed.
- Variability: Relations among variability in domain engineering, application engineering, and variability explicitly defined in variability model are managed through the variability trace links.

6 Variability tracing management

6.1 General

Variability tracing plan, enabling environments, and other managerial supports address variability traceability establishment and maintenance with variability model.

The variability tracing management process includes the following subprocesses:

- variability tracing planning;
- variability tracing enabling;
- variability tracing managing.

6.2 Variability tracing planning

6.2.1 Purpose of variability tracing planning

6.2.1.1 General

The purpose of this subprocess is to establish and maintain plans for defining and maintaining variability traceabilities in a product line.

6.2.1.2 Inputs

The following inputs should be available to perform the variability tracing planning process:

- variability management plan;
- organization's process capability;

- organization's available resources.

6.2.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the variability tracing planning process:

- variability tracing management plans are established;
- quality measures for variability tracing are defined;
- responsibilities for variability tracing are assigned.

6.2.1.4 Tasks

The organization shall implement the following tasks with respect to the variability tracing planning process.

- Design variability tracing strategy: to make a strategy for establishing and maintaining variability tracing centred on variability model.
- Define quality assurance measures for variability tracing: to define quality assurance measures to monitor and control quality assurance activities in variability tracing so as to achieve quality assurance throughout the SSPL stages.
- Assign responsibility for variability tracing: to clarify roles and responsibilities with respect to variability tracing.
- Record variability tracing plan: to document and get agreement and commitment on variability tracing plan from the appropriate staffs and managers.

6.2.2 Design variability tracing strategy

The goal of this task is to define the level of maintaining variability traceability, standards and methods used for establishing and maintaining variability traceability, rules, constraints, and validation related strategy for established variability traceability.

The method should support designing variability tracing strategy with the following capabilities:

- providing selection guides for variability tracing relevant standards, methods, validation;
- providing templates for informing what is major contents of strategy;
- providing examples for major documentation contents.

A tool should support designing variability tracing strategy by allowing the user to do the following:

- supporting the access of selection guides;
- providing electronic documentation template for strategy;
- allowing immediate reference for the examples of major documentation contents during documentation.

6.2.3 Define quality assurance measures for variability tracing

The goal of this task is to define quality assurance measures for quantifying the quality of variability tracing. Quality assurance measures include measures representing the quality level of established variability traceability, quality level of implemented quality assurance activities, and deviations from the defined rules and constraints.

The method should support defining quality assurance measures for variability tracing with the following capabilities:

- providing characteristic functions for expressing the overall quality level of variability traceability;
- supporting quality measure and metric definition;
- supporting quality measurement activities related to variability tracing.

A tool should support defining quality assurance measures for variability tracing by allowing the user to do the following:

- supporting calculation of characteristic functions;
- supporting data collection related to measures;
- allowing integration or expansion of quality measures for quality level analysis.

6.2.4 Assign responsibility for variability tracing

The goal of this task is to assign roles and responsibilities for variability tracing. Roles and responsibilities include identifying sources and targets of trace links, establishing variability trace links with tools, validating and verifying the established variability tracing, and quality assurance for variability tracing.

The method should support assigning responsibility for variability tracing with the following capabilities:

- extracting organization units that are proper to the defined variability tracing roles;
- assuring their capabilities with the roles and responsibilities to be assigned;
- providing a way to define roles and responsibilities concretely, correctly, and consistently for the right communication.

A tool should support assigning responsibility for variability tracing by allowing the user to do the following:

- allowing access for the structure of organization units;
- providing documentation for the assignment results;
- sharing the defined roles and responsibilities with relevant participants.

6.2.5 Record variability tracing plan

The goal of this task is to document plans for variability tracing. Plan documentation includes schedules, defined roles and responsibilities, defined quality assurance measures.

The method should support recording variability tracing plan with the following capabilities:

- supporting variability tracing relevant scheduling (establishment, management);
- supporting integration of the pre-defined roles and responsibilities to variability tracing plan;
- providing documentation template for variability tracing plan.

A tool should support recording variability tracing plan by allowing the user to do electronic plan documentation.

6.3 Variability tracing enabling

6.3.1 Purpose of variability tracing enabling

6.3.1.1 General

The purpose of this subprocess is to provide enabling environments for defining and maintaining variability traceabilities in a product line.

6.3.1.2 Inputs

The following inputs should be available to perform the variability tracing enabling process:

- variability tracing management plans;
- quality measures for variability tracing.

6.3.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the variability tracing enabling process:

- guidance for variability tracing is defined;
- roles and responsibilities for variability tracing are mobilized;
- variability tracing environments are enabled;
- quality assurance measurement environments for variability tracing are enabled.

6.3.1.4 Tasks

The organization shall implement the following tasks with respect to the variability tracing enabling process.

- Provide guidance for variability tracing: to define detailed ways to establish and maintain variability tracing centred on variability model.
- Mobilize roles and responsibilities for variability tracing: to provide resources appropriate for roles and responsibilities necessary to variability tracing.
- Enable variability tracing management: to provide managerial environments necessary to establishing and maintaining variability tracing centred on variability model.
- Enable variability tracing operations: to provide enablers necessary to right use of established variability tracing.
- Enable quality assurance measurement for variability tracing: to provide enablers necessary to measuring quality assurance activities.

6.3.2 Provide guidance for variability tracing

The goal of this task is to define detailed guides used for variability traceability establishment and management. The guidance includes detailed procedures, techniques used at the specific steps, detailed rules and constraints adhered during procedures, and required supports for the successful variability tracing in SSPL.

The method should support providing guidance for variability tracing with the following capabilities:

- defining the contents of the guidance;

- providing examples for each content.

A tool should support providing guidance for variability tracing by allowing the user to do the following:

- providing documentation environment for variability tracing guidance;
- allowing easy reference to each contents of guidance.

6.3.3 Mobilize roles and responsibilities for variability tracing

The goal of this task is to make all assigned roles and responsibilities ready for variability tracing.

The method should support mobilizing roles and responsibilities for variability tracing with the following capabilities:

- providing supports required for completing each role and responsibility's assignment;
- allowing early discovering of uncompleted roles and responsibilities.

A tool should support mobilizing roles & responsibilities for variability tracing by allowing the user to do the following:

- allowing traces for each role's current status;
- supporting share of the same views for the roles and responsibilities between managers and persons in charge.

6.3.4 Enable variability tracing management

The goal of this task is to provide guidance, tracing environments, and measurement environments for variability tracing management.

The method should support enabling variability tracing management with the following capabilities:

- providing ways to provide managerial support variability tracing management based on variability model;
- establishing managerial supports required for variability model-centred variability tracing;
- devising traceability management methods that are proper to support variability model-centred variability tracing management.

A tool should support enabling variability tracing management by allowing the user to do the following:

- implementing ways for variability model-centric variability tracing management;
- supporting enabling functions variability model-centric variability tracing management.

6.3.5 Enable variability tracing operations

The goal of this task is to make roles in charge of variability tracing operation ready for implementing established variability tracing strategy and plans.

The method should support enabling variability tracing operations with the following capabilities:

- supporting variability tracing algorithms enabling variability model-centric variability tracing management.

A tool should support enabling variability tracing operations by allowing the user to do the following:

- providing graphical notations to model variability;
- allowing variability model management.

6.3.6 Enable quality assurance measurement for variability tracing

The goal of this task is to make quality assurance roles ready for measuring the quality of variability tracing.

The method should support enabling quality assurance measurement for variability tracing with the following capabilities:

- providing enablers (i.e. enabling quality assurance roles, tools, methods) that make it possible to quality assurance measurement;
- providing enablers that make it possible to improve variability tracing quality based on quality assurance measurement results.

A tool should support enabling quality assurance measurement for variability tracing by allowing the user to support enablers that interact with each other.

6.4 Variability tracing managing

6.4.1 Purpose of variability tracing managing

6.4.1.1 General

The purpose of this subprocess is to provide managerial support for variability tracing.

6.4.1.2 Inputs

The following inputs should be available to perform the variability tracing managing process:

- variability tracing plans;
- actual data for variability tracing.

6.4.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the variability tracing managing process:

- issues to correct are derived;
- status of corrective actions for variability tracing is managed;
- inputs for improvement are generated.

6.4.1.4 Tasks

The organization shall implement the following tasks with respect to the variability tracing managing process.

- Review the plan versus actual result of variability tracing: to assign the quality objectives achieved through variability tracing with their pre-defined measures and metrics, and monitor whether they are operated in line with the pre-defined plans.
- Control issues on domain/application variability tracing: to examine issues raised during establishing and using variability tracing centred on variability model in domain/application engineering.
- Control issues on variability tracing management: to examine issues raised during variability tracing management.

- Control issues on variability tracing support: to examine issues raised during variability tracing support activities.
- Support corrective actions for variability tracing: to help repair issues in variability tracing.
- Make improvement actions for variability tracing: to improve variability traceability so as to achieve the assigned quality objectives.

6.4.2 Review the plan versus actual result of variability tracing

The goal of this task is to check the progress of variability tracing.

The method should support reviewing the plan versus actual result of variability tracing with the following capabilities:

- defining monitoring procedures for variability traceability establishment and management status compared to plans;
- providing ways to collect data for comparing plan versus actual variability tracing relevant operations;
- defining ways to integrate data for judging the status.

A tool should support reviewing the plan versus actual result of variability tracing by allowing the user to do the following:

- sharing consensus on the defined roles and responsibilities;
- providing (semi-) automated measurement environment for data collection and integration.

6.4.3 Control issues on domain/application variability tracing

The goal of this task is to collect and review issues raised during orthogonal variability model-centric domain and application variability traceability establishment.

The method should support controlling issues on domain/application variability tracing with the following capabilities:

- providing ways to find and review obstacles to the successful operation of variability tracing;
- defining decision criteria for classifying issues required for corrective actions;
- providing ways to defining corrective actions based on the type of obstacles reviewed in review issues of variability tracing task;
- providing documentation templates for recoding assessment results and corrective action plans.

A tool should support controlling issues on domain/application variability tracing by allowing the user to do the following:

- supporting issues collection raised by different roles and responsibility;
- allowing documentation for assessment results and corrective action plans.

6.4.4 Control issues on variability traceability management

The goal of this task is to collect and review issues raised during managing orthogonal variability model-centric variability traceability management.

The method should support controlling issues on variability traceability management with the following capabilities:

- providing ways to find and review obstacles to the successful operation of variability traceability management;
- defining decision criteria for classifying issues required for corrective actions;
- providing ways to defining corrective actions based on the type of obstacles reviewed in review issues of variability traceability management task;
- providing documentation templates for recoding assessment results and corrective action plans.

A tool should support controlling issues on variability traceability management by allowing the user to do the following:

- supporting issues collection raised by different roles and responsibility;
- allowing documentation for assessment results and corrective action plans.

6.4.5 Control issues on variability tracing support

The goal of this task is to collect and review issues raised on managerial supports and enablers for variability tracing.

The method should support controlling issues on variability tracing support with the following capabilities:

- providing ways to find and review obstacles to the successful operation of variability tracing support;
- defining decision criteria for classifying issues required for corrective actions;
- providing ways to defining corrective actions based on the type of obstacles reviewed in review issues of tasks related to variability tracing support;
- providing documentation templates for recoding assessment results and corrective action plans.

A tool should support controlling issues on variability tracing support by allowing the user to do the following:

- supporting issues collection raised by different roles and responsibility;
- allowing documentation for assessment results and corrective action plans.

6.4.6 Support corrective actions for variability tracing

The goal of this task is to implement corrective actions for resolving issues raised in variability tracing.

The method should support corrective actions for variability tracing with the following capabilities:

- providing ways to monitor and control the status of correction action;
- communicating corrective action results with the relevant participants.

A tool should support corrective actions for variability tracing by allowing the user to do the following:

- supporting simulation for confirming the achievement of quality level related to the variability tracing operation;
- allowing traces for the status of corrective actions.

6.4.7 Make improvement actions for variability tracing

The goal of this task is to define and implement improvement actions for variability traceability establishment, management, and supports.

The method should support making improvement actions for variability tracing with the following capabilities:

- collecting the improvement items of variability tracing operation;
- implementing the improvement items of variability tracing operation.

A tool should support making improvement actions for variability tracing by allowing the user to do the following:

- supporting improvement input collection;
- providing communication environment among the relevant participants.

7 Variability tracing

7.1 General

Variability of a product line should be traced from its introduction stage to binding stage. Because variability is implemented by using various mechanisms and it is represented by various forms in different development artefacts, variability tracing should support consistent tracing services.

The variability tracing process includes the following subprocesses:

- traceability between domain variability model and domain artefacts;
- traceability between application variability model and application artefacts;
- traceability between domain and application variability model;
- traceability between variability related artefacts within a stage;
- traceability between variability related artefacts in different adjacent stages.

NOTE [Annex A](#) provides example traceability dimension and [Annex B](#) provides several types of traceability relations. They can be referred to when decisions for establishing variability tracing are made.

7.2 Traceability between domain variability model and domain artefacts

7.2.1 Purpose of traceability between domain variability model and domain artefacts

7.2.1.1 General

The purpose of this subprocess is to establish explicit trace links between elements of domain variability model (variation point, variants) and relevant domain artefacts.

7.2.1.2 Inputs

The following inputs should be available to perform the traceability between domain variability model and domain artefacts process:

- domain variability models;
- domain artefacts representing variation point;

- domain artefacts realizing variants.

7.2.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the traceability between domain variability model and domain artefacts process:

- trace links from variation points in domain variability models to domain artefacts are established;
- trace links from variants in domain variability models to relevant domain artefacts are established.

7.2.1.4 Tasks

The organization shall implement the following tasks with respect to the traceability between domain variability model and domain artefacts process.

- Identify relations between variation point and domain artefact: to find domain artefacts that include the variation point.
- Identify relations between variant and domain artefact: to find domain artefacts that implement the variant.
- Define the trace links between domain variability model and domain artefacts: to establish trace links in accordance with the identified links.

7.2.2 Identify relations between variation point and domain artefact

The goal of this task is to place a domain artefact and its specific variation point.

The method should support identifying relations between variation point and domain artefact with the following capabilities:

- identifying the form of a variation point in accordance with the used variability mechanism;
- defining a way to establish trace link for “represented by” relation in accordance with the used variability mechanism.

A tool should support identifying relations between variation point and domain artefact by allowing the user to do the following:

- allowing the representation of a variation point in accordance with the used variability mechanism (in order to use for tracing);
- implementing the defined way to establish trace link for “represented by” relation in accordance with the used variability mechanism.

7.2.3 Identify relations between variant and domain artefact

The goal of this task is to place domain artefact that has “realize by” relation with a variant.

The method should support identifying relations between variant and domain artefact with the following capabilities:

- defining the form of a variant in accordance with the used variability mechanism;
- defining a way to establish trace link for “realized by” relation in accordance with the used variability mechanism.

A tool should support identifying relations between variant and domain artefact by allowing the user to do the following:

- allowing the representation of a variant in accordance with the used variability mechanism (in order to use for tracing);
- implementing the defined way to establish trace link for “realized by” relation in accordance with the used variability mechanism.

7.2.4 Define the trace links between domain variability model and domain artefacts

The goal of this task is to establish trace links between variability model orthogonally defined and domain artefacts.

The method should support defining the trace links between domain variability model and domain artefacts with the following capabilities:

- providing the representations for the sources and targets of trace links;
- enabling CRUD for trace links between domain variability model and domain artefacts;
- supporting explicit display (textual or graphical) for the established trace links;
- enabling trace link instantiation in accordance with variability binding in application engineering.

A tool should support defining the trace links between domain variability model and domain artefacts by allowing the user to do the following:

- implementing the defined representation for the sources and targets of trace links;
- implementing enablers for CRUD for trace links between domain variability model and domain artefacts;
- implementing the defined explicit display for the established trace links;
- supporting (semi-) automatic trace link instantiation in accordance with variability binding in application engineering.

7.3 Traceability between application variability model and application artefacts

7.3.1 Purpose of traceability between application variability model and application artefacts

7.3.1.1 General

The purpose of this subprocess is to establish explicit trace links between application variability model and application artefacts.

7.3.1.2 Inputs

The following inputs should be available to perform the traceability between application variability model and application artefacts process:

- application variability model;
- application artefacts related to selected variants;
- application-specific artefacts.

7.3.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the traceability between application variability model and application artefacts process:

- trace links from variants to relevant application artefacts are established;
- trace links from application-specific variability to application-specific artefacts are established.

7.3.1.4 Tasks

The organization shall implement the following tasks with respect to the traceability between application variability model and application artefacts process.

- Identify relations between variation point and application artefact: to find application artefacts that include the variation points.
- Identify relations between variant and application artefact: to find application artefacts that implement application specific variants.
- Define the trace links between application variability model and application artefacts: to establish trace links in accordance with the identified links.

7.3.2 Identify relations between variation point and application artefact

The goal of this task is to place an application artefact and its application specific variation point.

The method should support identifying relations between variation point and application artefact with the following capabilities:

- identifying the form of a variation point in accordance with the used variability mechanism;
- defining a way to establish trace link for ‘represented by’ relation in accordance with the used variability mechanism.

A tool should support identifying relations between variation point and application artefact by allowing the user to do the following:

- allowing the representation of a variation point in accordance with the used variability mechanism (in order to use for tracing);
- implementing the defined way to establish trace link for “represented by” relation in accordance with the used variability mechanism.

7.3.3 Identify relations between variant and application artefact

The goal of this task is to determine application specific artefacts that implement an application specific variant.

The method should support identifying relations between variant and application artefact with the following capabilities:

- defining the form of a variant in accordance with the used variability mechanism;
- supporting trace link establishment for “realized by” relation in accordance with the used variability mechanism.

A tool should support identifying relations between variant and application artefact by allowing the user to do the following:

- allowing the representation of a variant in accordance with the used variability mechanism (in order to use for tracing);

- implementing the defined way to establish trace link for “realized by” relation in accordance with the used variability mechanism.

7.3.4 Define the trace links between application variability model and application artefacts

The goal of this task is to establish trace links between variability model orthogonally defined and application artefacts.

The method should support defining the trace links between application variability model and application artefacts with the following capabilities:

- providing the representations for the sources and targets of trace links in application specific variability and relevant application artefacts;
- enabling CRUD for trace links between applications specifically added/adapted variability and application specific artefacts;
- supporting explicit display (textual or graphical) for the established trace links.

A tool should support defining the trace links between application variability model and application artefacts by allowing the user to do the following:

- implementing the defined representation for the sources and targets of trace links;
- implementing enablers for CRUD for trace links between applications specifically added/adapted variability and application specific artefacts;
- implementing the defined explicit display for the established trace links.

7.4 Traceability between domain and application variability model

7.4.1 Purpose of traceability between domain and application variability model

7.4.1.1 General

The purpose of this subprocess is to establish explicit trace links between domain variability model and bound variants in application variability model.

7.4.1.2 Inputs

The following inputs should be available to perform the traceability between domain and application variability model process:

- domain variability model;
- application variability model;
- binding results.

7.4.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the traceability between domain and application variability model process:

- trace links of domain variability model to application variability model are established.

7.4.1.4 Tasks

The organization shall implement the following tasks with respect to the traceability between domain and application variability model process.

- Identify relations between domain and application variability model: to find variability bindings in application variability model.
- Define trace links in accordance with binding results: to establish trace links in accordance with the identified variability bindings.

7.4.2 Identify relations between domain and application variability model

The goal of this task is to place relations between domain variability model and its resolution results.

The method should support identifying relations between domain and application variability model with the following capabilities:

- placing source and target objects between domain variability model and application variability model;
- supporting maintaining annotations and additional information defined between placed source and target objects.

A tool should support identifying relations between domain and application variability model by allowing the user to do the following:

- enabling reference for the elements of domain variability model and application variability model;
- extracting annotations and additional information together with the identified relations.

7.4.3 Define trace links in accordance with binding results

The goal of this task is to establish trace links between domain variability model and its application model.

The method should support defining trace links in accordance with binding results with the following capabilities:

- supporting seamless traces from domain variability model to application variability model in different lifecycle stages;
- enabling reference to added annotation for resolutions related to trace links.

A tool should support defining trace links in accordance with binding results by allowing the user to do the following:

- implementing seamless traces from domain variability model to application variability model in different lifecycle stages;
- providing enable/disable function to annotations added to application variability model with the selected trace link.

7.5 Traceability among variability related artefacts within a stage

7.5.1 Purpose of traceability between variability related artefacts within a stage

7.5.1.1 General

The purpose of this subprocess is to establish explicit trace links among variability related artefacts developed within the same stage. This traceability is established and managed through orthogonal variability model (i.e. domain variability model and application variability model).

7.5.1.2 Inputs

The following inputs should be available to perform the traceability between variability related artefacts within a stage process:

- domain variability models;
- application variability models;
- related variable domain artefacts within the same stage;
- bindings resolved during application engineering.

7.5.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the traceability between variability related artefacts within a stage process:

- trace links among variable domain artefacts within the same stage are established;
- instantiated trace links among instantiated application artefacts are derived.

7.5.1.4 Tasks

The organization shall implement the following tasks with respect to the traceability between variability related artefacts within a stage process.

- Identify relation among domain artefacts within the same stage: to find different artefacts that implement the same variability.
- Define trace links among domain artefacts within the same stage: to establish trace links among artefacts related to the same variability with variability model as the centre.
- Instantiate trace links in line with binding results: to establish trace links among relevant artefacts within the same stage by instantiating trace links established in domain engineering.

7.5.2 Identify relations among domain artefacts within the same stage

The goal of this task is to place variability relations represented/implemented by different domain artefacts.

The method should support identifying relations among domain artefacts within the same stage with the following capabilities:

- placing source and target objects in domain artefacts;
- supporting maintaining annotations and additional information defined between placed source and target objects.

A tool should support identifying relations among domain artefacts within the same stage by allowing the user to do the following:

- enabling reference for the elements of domain artefacts;
- extracting annotations and additional information together with the identified relations.

7.5.3 Define trace links among domain artefacts within the same stage

The goal of this task is to establish trace links in accordance with the identified relations.

The method should support defining trace links among domain artefacts within the same stage with the following capabilities:

- supporting seamless traces from sources of domain artefacts to targets of domain artefacts in the same lifecycle stages;
- enabling reference to added annotation for resolutions related to trace links.

A tool should support defining trace links among domain artefacts within the same stage by allowing the user to do the following:

- implementing seamless traces from sources of domain artefacts/domain artefacts' elements to targets of domain artefacts/domain artefacts' elements in the same lifecycle stages;
- providing enable/disable function to annotations added to the selected trace link.

7.5.4 Instantiate trace links in line with binding results

The goal of this task is to support trace link instantiation in accordance with binding results.

The method should support instantiating trace links in line with binding results with the following capabilities:

- supporting trace link instantiation with member product instantiation (or member product derivation from domain);
- maintaining instantiated trace links together with application variability model (in other words, resolution tree) for their easy change management.

A tool should support instantiating trace links in line with binding results by allowing the user to do the following:

- instantiating trace links (semi-) automatically together with member product derivation;
- synchronizing instantiated trace links with application variability model so that changes in variability can be reflected (semi-) automatically.

7.6 Traceability between variability related artefacts in adjacent stages

7.6.1 Purpose of traceability between variability related artefacts in different adjacent stages

7.6.1.1 General

The purpose of this subprocess is to establish explicit trace links between variability related artefacts developed in different adjacent stages.

7.6.1.2 Inputs

The following inputs should be available to perform the traceability between variability related artefacts in adjacent stages process:

- related variable domain artefacts in different adjacent stages;
- binding results during application engineering.

7.6.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the traceability between variability related artefacts in adjacent stages process:

- trace links among different related domain artefacts in adjacent stages are established;
- trace links for instantiated application artefacts are instantiated in accordance with binding results.

7.6.1.4 Tasks

The organization shall implement the following tasks with respect to the traceability between variability related artefacts in adjacent stages process.

- Identify relations between domain artefacts in adjacent stages: to find different artefacts that implement the same variability.
- Define trace links between domain artefacts in adjacent stages: to establish trace links among artefacts related to the same variability with variability model as the centre.
- Instantiate trace links in line with binding results: to establish trace links among the relevant artefacts of adjacent stages by instantiating trace links established in domain engineering.

7.6.2 Identify relations between domain artefacts in adjacent stages

The goal of this task is to place variability relations represented/implemented in different lifecycle stages.

The method should support identifying relations between domain artefacts in adjacent stages with the following capabilities:

- placing source and target objects between domain variability model and application variability model;
- supporting maintaining annotations and additional information defined between placed source and target objects.

A tool should support identifying relations between domain artefacts in adjacent stages by allowing the user to do the following:

- enabling reference for the elements of domain variability model and application variability model;
- extracting annotations and additional information together with the identified relations.

7.6.3 Define trace links between domain artefacts in adjacent stages

The goal of this task is to establish trace links in accordance with the identified relations.

The method should support defining trace links between domain artefacts in adjacent stages with the following capabilities:

- supporting seamless traces from sources of domain artefacts to targets of domain artefacts in the adjacent lifecycle stages;
- enabling reference to added annotation for resolutions related to trace links.

A tool should support defining trace links between domain artefacts in adjacent stages by allowing the user to do the following:

- implementing seamless traces from sources of domain artefacts to targets of domain artefacts in the adjacent lifecycle stages;
- providing enable/disable function to annotations added to the selected trace link.

7.6.4 Instantiate trace links in line with binding results

The goal of this task is to generate instantiations of trace links in accordance with binding results.

The method should support instantiating trace links in line with binding results with the following capabilities:

- supporting trace link instantiation with member product instantiation (or member product derivation from domain);
- maintaining instantiated trace links together with application variability model (in other words resolution tree) for their easy change management.

A tool should support instantiating trace links in line with binding results by allowing the user to do the following:

- instantiating trace links (semi-) automatically together with member product derivation;
- synchronizing instantiated trace links with application variability model so that changes in variability can be reflected (semi-) automatically.

8 Variability tracing support

8.1 General

For constructing complete and consistent variability tracing, end-to-end variability traceability should be realized. Variability tracing support covers the remaining traceability that is necessary but is not covered by the variability tracing process and traceability algorithm necessary to support end-to-end variability traceability. And this process helps ensure the quality of activities and work products conducted/produced during variability tracing activities.

The variability tracing support process includes the following subprocesses:

- relating traceability to variability model;
- traceability attribute;
- traceability algorithm;
- quality assurance for variability tracing.

8.2 Relating tracing to variability model

8.2.1 Purpose of relating tracing to variability model

8.2.1.1 General

The purpose of this subprocess is to enable establishing trace links for the rest of the aspects of variability (e.g. variability mechanism or others not covered in [Clause 7](#)) with variability model.

8.2.1.2 Inputs

The following inputs should be available to perform the relating tracing to variability model process:

- variability traceabilities established and maintained.

8.2.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the relating tracing to variability model process:

- refined variability traceabilities are established and maintained.

8.2.1.4 Tasks

The organization shall implement the following tasks with respect to the relating tracing to variability model process.

- Identify variability tracing conflicts: to find omitted variability tracing or conflicts with overall product line traceability.
- Harmonize variability tracing with product line traceability: to realize the seamless traceability of overall product line traceability by refining variability tracing.

8.2.2 Identify variability tracing conflicts

The goal of this task is to identify variability-related traceability that is not properly established and managed.

The method should support identifying variability tracing conflicts with the following capabilities:

- providing variability tracing framework consisting of the whole required variability tracing;
- supporting decision making for identifying variability tracing conflicts.

A tool should support identifying variability tracing conflicts by allowing the user to do the following:

- implementing variability tracing framework;
- providing immediate recommendation for identifying variability tracing conflicts.

8.2.3 Harmonize variability traceability with product line traceability

The goal of this task is to coordinate the established variability traceability with that in the overall product line.

The method should support harmonizing variability traceability with product line traceability with the following capabilities:

- providing product line traceability for comparing with the established variability traceability;
- supporting comparative exploration with product line traceability.

A tool should support harmonizing variability traceability with product line traceability by allowing the user to do the following:

- allowing reference for the defined product line traceability;
- providing environment for comparative exploration with product line traceability.

8.3 Quality assurance for variability tracing

8.3.1 Purpose of quality assurance for variability tracing

8.3.1.1 General

The purpose of this subprocess is to help ensure that variability tracing and their relevant processes comply with predefined provisions and plans.

8.3.1.2 Inputs

The following inputs should be available to perform the quality assurance for variability tracing process:

- trace links established in a product line;
- processes used for variability tracing;
- quality assurance measures defined in variability tracing planning.

8.3.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the quality assurance for variability tracing process:

- variability tracing-specific quality assurance strategy is developed;
- evidences of variability tracing quality assurance are produced and maintained;
- non-conformance issues in variability tracing are identified and recorded;
- adherence of variability tracing and variability tracing processes to the PL organizationally adopted standards, rules, and constraints are assured.

8.3.1.4 Tasks

The organization shall implement the following tasks with respect to the quality assurance for variability tracing process.

- Objectively evaluate variability tracing activities: to help ensure that employed activities for variability tracing comply with the provisions and plans.
- Objectively evaluate variability tracing work products: to help ensure that produced planned artefacts satisfy the predefined quality criteria.
- Communicate and resolve noncompliance issues: to help ensure that the noncompliance issues of variability tracing are objectively tracked, communicated, and resolved with appropriate staffs and managers.
- Establish records of tracing quality assurance activities: to record and revise quality assurance activities performed with respect to variability tracing.

8.3.2 Objectively evaluate variability tracing activities

The goal of this task is to assure whether the implemented variability tracing activities adhere to the defined processes.

The method should support objectively evaluating variability tracing activities with the following capabilities:

- assuring employed variability traceability establishment and management activities;
- assuring performed variability traceability establishment and management practices;
- assuring measurement processes related to variability tracing;
- assuring activities related to variability tracing supports (e.g. conflict mitigation, harmonization with product line traceability).

A tool should support objectively evaluating variability tracing activities by allowing the user to do the following:

- supporting the access of PL organization's process definitions related to variability tracing (maintained in the organization's process repository);
- accessing performed variability tracing practices for quality assurance;
- allowing the access of performed measurement activities performed for variability traceability establishment and management.

8.3.3 Objectively evaluate variability tracing work products

The goal of this task is to assure the produced planned work products during variability traceability establishment and management.

The method should support objectively evaluating variability tracing work products with the following capabilities:

- assuring variability tracing work products (e.g. traceability map);
- assuring relations that they are correct, complete, and consistent with other relevant relations established from variability traceability perspective;
- assuring the work products of variability tracing supports;
- assuring provided variability tracing supports (e.g. tools, methods, resources).

A tool should support objectively evaluating variability tracing work products by allowing the user to do the following:

- allowing the access of variability tracing work products (e.g. established and maintained traceability map);
- allowing relation tracking for established trace links (e.g. traceability table or explicit link to trace);
- supporting the access of variability tracing support work products;
- providing a way to measure the satisfaction degree of the provided supports for variability traceability establishment and management.

8.3.4 Communicate and resolve noncompliance issues

The goal of this task is to define action items to resolve noncompliance issues found with relevant participants.

The method should support communicating and resolving noncompliance issues with the following capabilities:

- establish escalation lines for resolving noncompliance issues when they cannot be resolved with the appropriate variability tracing staffs (the escalation line could include variability tracing staffs, appropriate domain engineers, or application engineers);
- tracking noncompliance issues throughout the established escalation lines;
- analysing noncompliance issues if there are any variability relevant quality trends (i.e. right implementation, right binding or reuse with low cost in application engineering).

A tool should support communicating and resolving noncompliance issues by allowing the user to do the following:

- supporting documentation for noncompliance issues to be escalated;
- providing communication space among staffs or managers within the established escalation lines;
- sharing the status of noncompliance issues among staffs and managers within the established escalation lines
- supporting statistical analysis for variability relevant quality trends.

8.3.5 Establish records of tracing quality assurance activities

The goal of this task is to document quality assurance activities with their outcomes.

The method should support establishing records of tracing quality assurance activities with recording and revising variability tracing quality assurance activities.

A tool should support establishing records of traceability quality assurance activities by allowing the user to do the supporting documentation for variability tracing quality assurance activities.

8.4 Variability tracing algorithm

8.4.1 Purpose of variability tracing algorithm

8.4.1.1 General

The purpose of this subprocess is to define and implement algorithms essential to establish and maintain variability traceability with different abstraction levels and around orthogonal variability model.

8.4.1.2 Inputs

The following inputs should be available to perform the variability tracing algorithm process:

- variability tracing plans;
- algorithms used to variability tracing.

8.4.1.3 Outcomes

The following outcomes shall be available as a result of the successful implementation of the variability tracing algorithm process:

- variability tracing algorithms are implemented;
- the performance of algorithms is measured;
- optimized variability tracing algorithms are produced.

8.4.1.4 Tasks

The organization shall implement the following tasks with respect to the variability tracing algorithm process.

- Identify algorithms for variability tracing: to find or develop appropriate algorithms used for establishing and maintaining variability traceability.
- Evaluate defined variability tracing algorithms: to review the selected or developed algorithms' capability with respect to the defined level of variability traceability.
- Measure the performance of algorithms: to keep track of the selected or developed algorithms' capability.
- Optimize traceability algorithms: to resolve issues specified as the results of measurement.

8.4.2 Identify algorithms for variability tracing

The goal of this task is to search, acquire, or develop algorithms required for establishing, maintaining, or querying variability tracing.

The method should support identifying algorithms for variability tracing with the following capabilities:

- defining algorithms used for querying variability tracing by relation types (detailed relation types are described in [Annex C](#));
- defining algorithms used for analysing impacts on variability relevant domain artefacts by tracing the established variability traceability;
- defining algorithms used for analysing impacts on variability relevant application artefacts by tracing the established variability traceability.

A tool should support identifying algorithms for variability tracing by allowing the user to do the following:

- implementing algorithms used for querying variability tracing by relation types;
- implementing algorithms used for analysing impacts on variability relevant domain artefacts by tracing the established variability traceability;
- implementing algorithms used for analysing impacts on variability relevant application artefacts by tracing the established variability tracing.

8.4.3 Evaluate defined variability tracing algorithms

The goal of this task is to assess the effectiveness and efficiency of the defined variability tracing algorithms.

The method should support evaluating defined variability tracing algorithms with the following capabilities:

- assessing algorithms against their usage purposes;
- assessing the consistency supports of algorithms by traceability dimensions.

A tool should support evaluating defined variability tracing algorithms by allowing the user to do the following:

- supporting the simulation of algorithm usage;
- providing (semi-) automatic consistency checking of algorithms by traceability dimensions;
- providing experiments data for experimenting variability tracing algorithms.

8.4.4 Measure the performance of algorithms

The goal of this task is to provide quantitative measurement values for the performance of algorithms.

The method should support measuring the performance of algorithms with the following capabilities:

- providing measures for measuring the performance of algorithms quantitatively;
- providing metrics for representing the degree of the performance of algorithms quantitatively.

A tool should support measuring the performance of algorithms by allowing the user to do the following:

- supporting (semi-) automatic data collection;
- visualizing the degree of the performance explicitly.

8.4.5 Optimize traceability algorithms

The goal of this task is to improve variability tracing algorithms.

The method should support optimizing traceability algorithms with the following capabilities:

- defining improvement items for algorithms;
- assuring consistencies for each improvement item against traceability dimensions.

A tool should support optimizing traceability algorithms by allowing the user to do the following:

- sharing improvement items for algorithms among participants;
- supporting (semi-) automatic consistency check;
- visualizing the optimization points of traceability algorithms.

Annex A

(informative)

Variability traceability dimension

For reducing the complexity of variability traceability management, a coherent, orthogonal, and complete traceability dimensions should be defined. [Table A.1](#) describes the example orthogonal traceability dimensions.

Table A.1 — Traceability dimensions

| Dimensions | | Descriptions |
|--|-------------------------|---|
| Inter-stage | Refinement | Variability tracing between different life cycle stages. |
| Intra-stage | Similarity | Variability tracing within the same life cycle stages. |
| Variability model/ Domain artefacts | Variability/Realization | Traceability from variability model to development artefacts — representation relationship from a variation point to development artefacts; — “realize by” relationship from variants to development artefacts. |
| Domain variability model/Application variability model | Variability/Use | Variability tracing from domain variability to variability selection in a member product. |
| Versioning | | Traceability between the different versions of the same product line artefact (artefacts include variability model, analysis model, components). |

Annex B (informative)

Types of variability traceability relations

Depending on the traceability relation type, tracing ways for analysing change impacts can differ. For example, a dependency relation between e1 (source element) and e2 (target element) in a trace link means that target element e2 can exist only when source element e1 exists. In this case, when e1 has changed, it is important to analyse the impacts on e2. This means that when e2 has changed, the possibility of change impacts on e1 is low. In addition, the relation type can be a way to represent the level of abstraction of traceability. For example, in the case that the refinement results e1 are e2 and e3 and thereafter e2 is refined into e4 and e5, sources of trace links of level 0 can be extracted by using annotated relation types. In this way, traceability information of the specific level of abstraction can be represented. This resolves the complexity of traceability management and scalability problems due to many trace links being established. Traceability relation types can include the following.

- Dependency relation: Source element in a trace link requires a target element.
- Refinement relation: Target element in a trace link is a result of refinement into the detailed level. “LightManager” class has the refinement relation with “LightManager” component. So does “FrontManager” component and “FrontManager” class.
- Inclusion relation: Target element in a trace link is included in source element.
- Domain similarity relation: Target element in a trace link is another representation or implementation for the source element in domain engineering.
- Application similarity relation: Target element in a trace link is another representation or implementation for the source element in application engineering.
- Instantiation relation: Target element in a trace link is a binding result of source element. Trace links between “LightManager” class in domain class diagram and “LightManager” class in application class diagram and between “FrontManager” class in domain class diagram and “FrontManager” class in application class diagram are examples of this relation type. See [Figure B.1](#).

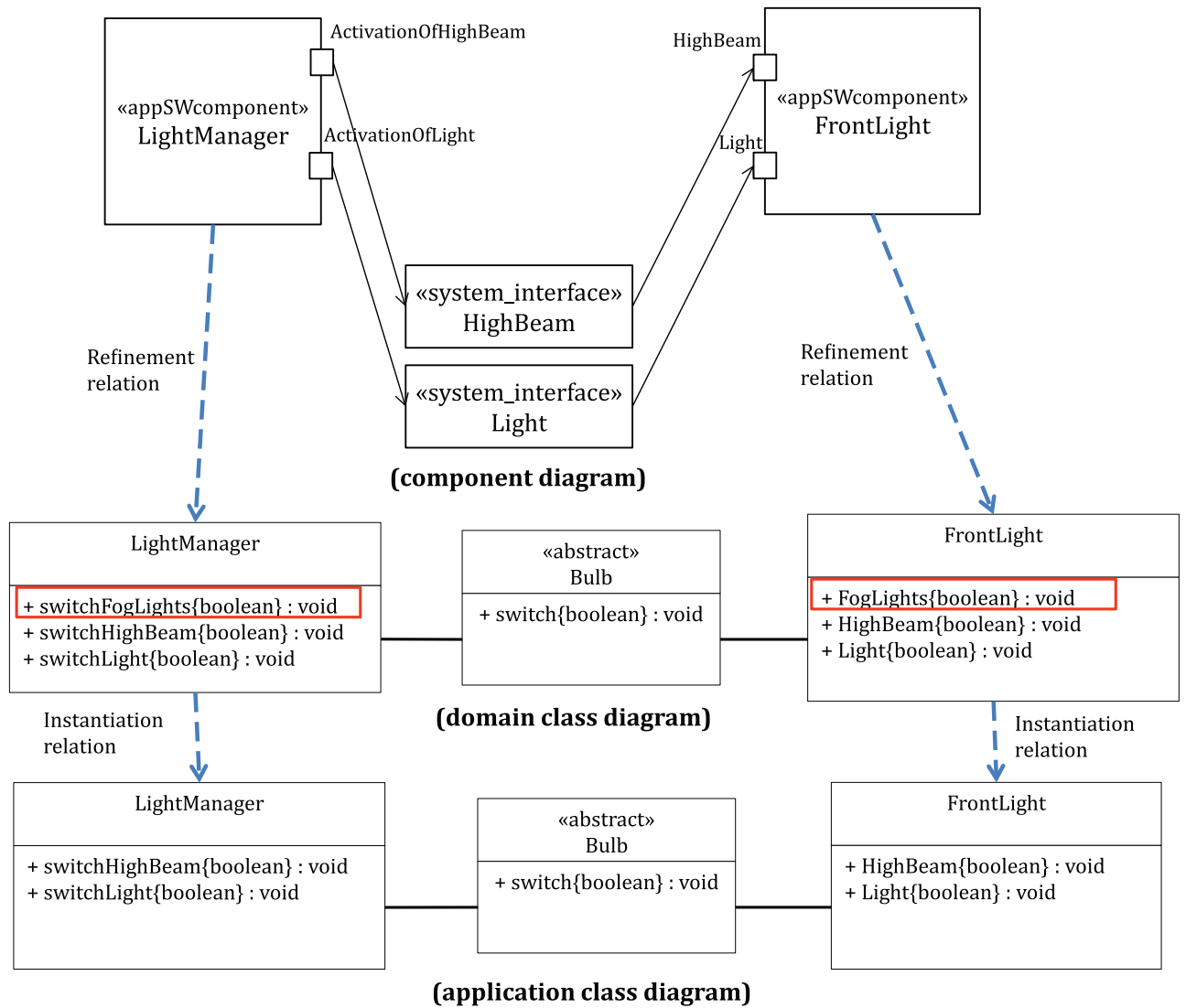


Figure B.1 — Example of relation types

Annex C (informative)

Variability traceability attributes

Variability traceability attributes include the following.

- Traceability dimensions: Refinement, Similarity, Variability/Realization, Variability/Use, Versioning (refer [Annex A](#)).
- Requirements traceability types: Functional or non-functional requirements relevant traceability.
- Traceability relation types: relationships between source and target tracing elements, e.g. dependency, refinement, inclusion, domain similarity, application similarity, instantiation (refer [Annex B](#)).
- Traceability levels: level of details for established traceability, e.g. the finer level of detail, the more general level. Refinement relation type can cover this attribute partially.
- Trace link semantics: Providing information that makes such the trace links be used in different ways. The value of this attribute can be shallow or richer semantics.

Bibliography

- [1] ISO/IEC 14102, *Information technology — Guideline for the evaluation and selection of CASE tools*
- [2] ISO/IEC 15940, *Systems and software engineering — Software Engineering Environment Services*
- [3] ISO/IEC/TR 19759, *Software Engineering — Guide to the software engineering body of knowledge (SWEBOK)*
- [4] ISO/IEC 25000, *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE*
- [5] POHL K., BÖCKLE G., VAN DER LINDEN F.J. *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer, 2005
- [6] PAUL C. *A Framework for Software Product Line Practice, Version 5.0*. Software Engineering Institute, Carnegie Mellon University, July 2007

