INTERNATIONAL STANDARD

ISO/IEC 26553

First edition
2018-11

Information technology — Software and systems engineering — Tools and methods for product line realization

Technologies de l'information — Ingénierie des systèmes et du logiciel — Outils et méthodes pour la réalisation d'une gamme de produits

Reference number
ISO/IEC 26553:2018(E)

© ISO/IEC 2018

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2018

All rights reserved. Unless otherwise specified, or required in the context of its implementation, 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
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword vi

Introduction vii

1 Scope 1

2 Normative references 1

3 Terms and definitions 1

4 Reference model for product line realization 3
4.1 Overview 3
4.2 Organization realization management 4
4.3 Domain realization 4
4.4 Domain implementation 5
4.5 Asset management in realization 5
4.6 Detailed application application design 6
4.7 Application implementation 6
4.8 Variability management in realization 6

5 Organizational realization management 8
5.1 General 8
5.2 Organizational planning for realization 8
5.2.1 Principal constituents 8
5.2.2 Confirm the readiness of realization 9
5.2.3 Define realization plans 10
5.3 Organizational enabling environment for realization 10
5.3.1 Principal constituents 10
5.3.2 Analyse requirements for enabling environments 11
5.3.3 Establish and maintain enabling environments 11
5.3.4 Enable interoperability among related infrastructure elements 11
5.4 Organizational operational managing for realization 12
5.4.1 Principal constituents 12
5.4.2 Monitor and control progress in realization 13
5.4.3 Make corrective action and improvement in realization 13

6 Detailed domain design 14
6.1 General 14
6.2 Detailed domain design initiation 14
6.2.1 Principal constituents 14
6.2.2 Review COTS from detailed design perspective 15
6.2.3 Confirm inputs for detailed domain design 15
6.2.4 Confirm detailed domain design capability 15
6.3 Detailed domain interface design 16
6.3.1 Principal constituents 16
6.3.2 Examine interactions among domain components 16
6.3.3 Define the detailed internal structures of domain interfaces 17
6.3.4 Verify and validate detailed domain interface design 17
6.3.5 Document detailed domain interface design 17
6.4 Detailed domain component design 18
6.4.1 Principal constituents 18
6.4.2 Define the detailed internal structures of domain components 19
6.4.3 Verify and validate detailed domain component design 19
6.4.4 Prepare test inputs for unit testing 19
6.4.5 Document detailed domain component design 20
6.5 Detailed software domain artefact design 20
6.5.1 Principal constituents 20
6.5.2 Define Detailed software domain artefact design 21
6.5.3 Verify and validate detailed software domain artefact design 21

6.5.4 Prepare test inputs for unit testing 22
6.5.5 Document detailed software domain artefact design 22
7 Domain implementation 22
7.1 General 22
7.2 Detailed domain implementation initiation 23
7.2.1 Principal constituents 23
7.2.2 Confirm inputs for domain implementation 23
7.2.3 Confirm domain implementation capability 24
7.3 Domain interface implementation 24
7.3.1 Principal constituents 24
7.3.2 Implement domain interface 24
7.3.3 Build domain interfaces 25
7.3.4 Verify and validate domain interface implementation 25
7.4 Domain component implementation 25
7.4.1 Principal constituents 25
7.4.2 Implement domain components 26
7.4.3 Build domain components 26
7.4.4 Verify and validate domain component implementation 27
7.4.5 Integrate domain components 27
7.5 Software domain artefact implementation 27
7.5.1 Principal constituents 27
7.5.2 Implement software domain artefacts 28
7.5.3 Build software domain artefacts 28
7.5.4 Verify and validate software domain artefacts 29
8 Variability management in realization 29
8.1 General 29
8.2 Variability mechanism category in realization 29
8.2.1 Principal constituents 29
8.2.2 Identify variability mechanisms in realization by category 31
8.2.3 Guide the use of variability mechanism category in realization 31
8.2.4 Trace the usage status of variability mechanism category in realization 31
8.2.5 Update variability mechanism category in realization 32
8.3 Variability in realization 32
8.3.1 Principal constituents 32
8.3.2 Model variability in realization 33
8.3.3 Maintain variability mechanisms in realization 34
8.3.4 Document variability in realization 34
8.4 Traceability of variability in realization 34
8.4.1 Principal constituents 34
8.4.2 Define trace links among variability in different realization artefacts 35
8.4.3 Define trace links between realization artefacts and variability model 35
9 Asset management in realization 36
9.1 General 36
9.2 Detailed domain design artefacts as domain assets 36
9.2.1 Principal constituents 36
9.2.2 Identify detailed design artefacts managed as domain assets 37
9.2.3 Define configuration and annotation in detailed domain design 37
9.3 Domain implementation artefacts as domain assets 38
9.3.1 Principal constituents 38
9.3.2 Identify domain implementation artefacts managed as domain assets 38
9.3.3 Define configuration and annotation in domain implementation 39
9.4 Attached process for reusing domain realization assets 39
9.4.1 Principal constituents 39
9.4.2 Identify processes adhered for realization asset reuse 40
9.4.3 Make attached process as a part of domain realization assets 40
9.5 Detailed application design artefacts as application assets 40
9.5.1 Principal constituents 40

© ISO/IEC 2018 – All rights reserved

9.5.2 Identify detailed application design artefacts managed as application assets 41
9.5.3 Define configuration and annotation in detailed application design 41
9.6 Application implementation artefacts as application assets 42
9.6.1 Principal constituents 42
9.6.2 Identify application implementation artefacts as application assets 42
9.6.3 Define configuration and annotation of application implementation 42

10 Detailed application design 43
10.1 General 43
10.2 Detailed application design initiation 43
10.2.1 Principal constituents 43
10.2.2 Derive detailed application design from detailed domain design 44
10.2.3 Validate derived detailed application design 44
10.2.4 Confirm detailed application design capability 45
10.3 Detailed application interface design 45
10.3.1 Principal constituents 45
10.3.2 Examine interactions among application components 46
10.3.3 Define the detailed internal structures of application interfaces 46
10.3.4 Verify and validate detailed application interface design 46
10.3.5 Document detailed application interface design 47
10.4 Detailed application component design 47
10.4.1 Principal constituents 47
10.4.2 Identify, evaluate and select COTS 48
10.4.3 Define the detailed internal structures of application components 48
10.4.4 Verify and validate detailed application component design 49
10.4.5 Document detailed application component design 49
10.5 Detailed software application artefact design 49
10.5.1 Principal constituents 49
10.5.2 Define the detailed internal structures of software application artefacts 50
10.5.3 Verify and validate detailed software application artefact design 50
10.5.4 Document the detailed design of software application artefacts 51

11 Application implementation 51
11.1 General 51
11.2 Application implementation initiation 51
11.2.1 Principal constituents 51
11.2.2 Derive application implementation from domain implementation 52
11.2.3 Validate derived application implementation 52
11.2.4 Confirm application implementation capability 53
11.3 Application interface implementation 53
11.3.1 Principal constituents 53
11.3.2 Implement the application interfaces 53
11.3.3 Build application interfaces 54
11.3.4 Verify and validate application interface implementation 54
11.4 Application component implementation 54
11.4.1 Principal constituents 54
11.4.2 Implement application components 55
11.4.3 Build application components 55
11.4.4 Verify and validate application component implementation 56
11.4.5 Integrate application components 56
11.5 Software application artefact implementation 56
11.5.1 Principal constituents 56
11.5.2 Implement software application artefacts 57
11.5.3 Build software application artefacts 58
11.5.4 Verify and validate software application artefact implementation 58
11.5.5 Integrate software application artefacts 58

Annex A (informative) **Scope of realization activities 59**

Bibliography 60

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 of 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 www.iso.org/iso/foreword.html.

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

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

© ISO/IEC 2018 – All rights reserved

Introduction

The main purpose of this document is to deal with the capabilities of methods and tools of software and systems product line (SSPL) realization which includes detailed design and implementation. This document defines how the tools and methods can support the software and systems product line-specific realization processes.

Domain realization will be carried out based on domain architecture that provides structures and constraints that govern the subsequent SSPL lifecycle processes. The outcomes of domain realization processes are transferred into the realization of a member product at the application realization processes. Therefore realization support tools and methods should consider both engineering processes, namely domain realization and application realization.

Product line realization can be differentiated from a single product development because of the following aspects:

— The outcomes of domain requirements engineering and domain architecture form the basis for product line realization unlike the case of a single product development. There are two core processes in product line realization: domain realization and application realization. The major aims of the domain realization processes are to conduct detailed design and further implementation based on domain architecture, which includes commonality and variability for a family of products, and to prepare necessary variability information for variability modelling. Whereas, the major aims of the application realization processes are to conduct detailed design and implementation for application realization and to bind variability whose defined binding time is realization stage.

This document can be used in the following modes:

— by the users of this document: to benefit people who conduct detailed design and implementation for software and systems product lines;

— by a product line organization: to provide guidance on the evaluation and selection for methods and tools for product line realization; and

— by providers of methods and tools: to provide guidance on implementing or developing tools and methods by providing a comprehensive set of the capabilities of tools and methods for product line realization.

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 product line realization. Other standards in the ISO/IEC 26550 family are as follows:

ISO/IEC 26550, ISO/IEC 26551, ISO/IEC 26555, ISO/IEC 26557, ISO/IEC 26558 and ISO/IEC 26559 are published. ISO/IEC 26552, ISO/IEC 26554, ISO/IEC 26556, ISO/IEC 26560, ISO/IEC 26561, ISO/IEC 26562 and ISO/IEC 26563 are planned International Standards. The following list provides an overview of the series:

— 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 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 modeling 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;
- 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; and
- others (ISO/IEC 26564 to ISO/IEC 26599) are to be developed.

© ISO/IEC 2018 – All rights reserved

Information technology — Software and systems engineering — Tools and methods for product line realization

1 Scope

This document, within the context of tools and methods of detailed design and implementation for software and system product lines:

— provides the terms and definitions specific to realization for software and systems product lines;

— defines processes performed during product line realization (those processes are described in terms of purpose, inputs, tasks and outcomes);

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

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

This document concerns processes and capabilities of realization tools and methods for a family of products, not for a single system.

2 Normative references

There are no normative references in this document.

© ISO/IEC 2018 – All rights reserved