INTERNATIONAL STANDARD

ISO/IEC 26580

First edition
2021-04

Software and systems engineering — Methods and tools for the feature-based approach to software and systems product line engineering

Ingénierie du logiciel et des systèmes — Méthodes et outils pour l'approche basée sur les caractéristiques dans l'ingénierie de lignes de produits logiciels et systèmes

Reference number
ISO/IEC 26580:2021(E)

© ISO/IEC 2021



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

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
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 ... **2**
4 Overview of feature-based product line engineering ... **4**
4.1 General ... 4
4.2 Shared assets ... 5
4.3 Features ... 6
4.4 Automated means of production ... 7
5 A feature-based specialization of software and systems product line engineering ... **7**
6 Reference model for the feature-based approach to software and systems product line engineering ... **11**
6.1 General ... 11
6.2 Key elements of the feature-based PLE factory ... 12
6.2.1 General ... 12
6.2.2 Feature catalogue ... 13
6.2.3 Bill-of-features and bill-of-features portfolio ... 13
6.2.4 Shared asset supersets ... 14
6.2.5 PLE factory configurator ... 14
6.2.6 Product asset instances ... 14
6.3 Relationships among the key elements of the factory ... 15
6.3.1 General ... 15
6.3.2 Feature-based abstractions: feature catalogue and bill-of-features portfolio ... 15
6.3.3 Domain supersets: feature catalogue and shared asset supersets ... 16
6.3.4 Assets: shared asset supersets and product asset instances ... 17
6.3.5 Product instances: bill-of-features portfolio and product asset instances ... 18
6.4 Reference model layers ... 19
6.5 Feature language ... 20
6.6 Support for a hierarchical product line of product lines ... 20
6.7 Other concerns ... 21
6.7.1 General ... 21
6.7.2 Configuration management concern ... 21
6.7.3 Traceability concern ... 21
6.7.4 Change management concern ... 22
6.7.5 Access control concern ... 22
7 Technology layer ... **22**
7.1 General ... 22
7.2 Feature language ... 22
7.3 Feature catalogue ... 25
7.4 Bill-of-features portfolio ... 25
7.5 Shared asset supersets ... 25
7.6 Product asset instances ... 26
7.7 PLE factory configurator ... 26
7.8 PLE factory development environment ... 26
8 Technical organization management layer ... **27**
8.1 General ... 27
8.2 Relationship to ISO/IEC 26550 technical management process group and ISO/IEC 26556 ... 28
8.3 Feature catalogue engineering ... 29
8.3.1 Purpose ... 29

© ISO/IEC 2021 – All rights reserved

8.3.2 Role ... 29
8.3.3 Outcomes ... 29
8.3.4 Inputs ... 29
8.3.5 Tasks ... 29
8.3.6 Tools ... 30
8.4 Bill-of-features portfolio engineering ... 31
8.4.1 Purpose ... 31
8.4.2 Role ... 31
8.4.3 Outcomes ... 31
8.4.4 Inputs ... 31
8.4.5 Tasks ... 31
8.4.6 Tools ... 32
8.5 Shared asset superset engineering ... 32
8.5.1 Purpose ... 32
8.5.2 Role ... 33
8.5.3 Outcomes ... 33
8.5.4 Inputs ... 33
8.5.5 Tasks ... 33
8.5.6 Tools ... 34
8.6 Automated configuration of the product asset instances ... 35
8.6.1 Purpose ... 35
8.6.2 Role ... 35
8.6.3 Outcomes ... 35
8.6.4 Inputs ... 35
8.6.5 Task — Configure the shared asset supersets using the PLE factory configurator ... 35
8.6.6 Tools ... 35
8.7 Verification, validation, and product delivery of the product asset instances ... 35
8.7.1 Purpose ... 35
8.7.2 Role ... 36
8.7.3 Outcomes ... 36
8.7.4 Inputs ... 36
8.7.5 Tasks ... 36
8.7.6 Tools ... 36
8.8 Configuration management ... 37
8.8.1 Purpose ... 37
8.8.2 Role ... 37
8.8.3 Outcomes ... 37
8.8.4 Inputs ... 37
8.8.5 Tasks ... 37
8.8.6 Tools ... 38
8.9 Traceability management ... 38
8.9.1 Purpose ... 38
8.9.2 Role ... 38
8.9.3 Outcomes ... 38
8.9.4 Inputs ... 39
8.9.5 Tasks ... 39
8.9.6 Tools ... 39
8.10 Change management ... 40
8.10.1 Purpose ... 40
8.10.2 Role ... 40
8.10.3 Outcomes ... 40
8.10.4 Inputs ... 40
8.10.5 Tasks ... 40
8.10.6 Tools ... 41

9 Business organization management layer ... 41
9.1 General ... 41
9.2 Incorporation of ISO/IEC 26550, ISO/IEC 26556 and ISO/IEC 26562 processes ... 42

© ISO/IEC 2021 – All rights reserved

9.3 Fund the PLE factory 44
9.3.1 Purpose 44
9.3.2 Outcomes 44
9.3.3 Inputs 44
9.3.4 Task — Establish and execute a funding policy for the PLE factory 44
9.3.5 Tools 45
Annex A (informative) **Terminology specialization from ISO/IEC 26550 to this document** **46**
Annex B (informative) **UML 2.0 Diagrams for the feature-based PLE factory** **50**
Bibliography **51**

© ISO/IEC 2021 – All rights reserved

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

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 or www.iec.ch/members_experts/refdocs).

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) or the IEC list of patent declarations received (see patents.iec.ch).

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. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint 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 and www.iec.ch/national-committees.

© ISO/IEC 2021 – All rights reserved

Introduction

Feature-based software and systems product line engineering ("feature-based PLE" for short) is a specialization of software and systems product line (SSPL) engineering and management that is described in ISO/IEC 26550. ISO/IEC 26550 describes a very generalized approach to SSPL, focusing on the benefits of exploiting a common platform of reusable assets for a product family. Each organization that adopts SSPL under ISO/IEC 26550 is free to do so using their preferred techniques and methods.

What is the motivation for creating a standard for a specialization of SSPL? As the SSPL field has matured and achieved widespread attention in the industry, a specific and repeatable approach to SSPL has emerged that takes advantage of commercial off-the-shelf industrial-strength tools and technology, along with robust best practices for methods and processes, that automate and formalize many of the processes in domain and application engineering. The result is that less upfront analysis, design, and implementation effort is required prior to gaining the benefits from the approach.

While SSPL in general provides significant benefits, it also requires a significant investment of time and effort to adopt and to ultimately achieve those benefits. The feature-based PLE specialization is a more narrowly defined solution that can be supported by off-the-shelf tools and methods, which has resulted in lower investments when an organization adopts SSPL. Feature-based PLE embodies lessons learned about SSPL practices that have been shown to provide some of the highest benefits and returns (see, for example, References [2] and[8]).

This document provides a reference model consisting of an abstract representation of the key technical elements, tools, and methods of feature-based PLE. The predominant specializations of general SSPL that characterize feature-based PLE are:

a) a mapping from features to asset variation points that is sufficient to drive a fully automated configurator that produces assets specific to member products;

b) a methodological shift of all design and implementation effort, change management, and configuration management to domain engineering, so that application engineering is reduced to automated configuration of member product instances and testing of configured member products and member-product-specific assets.

This document embodies a distinct separation of concerns between the feature-based PLE technology providers and feature-based technology users. For each of these stakeholder concerns, the scope of this document is to define only what is necessary and sufficient to enable feature-based PLE practice. For technology providers, this imparts flexibility in how the necessary and sufficient technical capabilities are provided, as well as the opportunity to offer more expansive capabilities that are possible in an ideal solution. For technology users, this provides flexibility to select among the technology providers and to apply the methods that best match their technical and business objectives for feature-based PLE.

© ISO/IEC 2021 – All rights reserved

Software and systems engineering — Methods and tools for the feature-based approach to software and systems product line engineering

1 Scope

This document is a specialization of the more general reference model for software and systems product line engineering and management described in ISO/IEC 26550. The specialization defined herein addresses a class of methods and tools referred to as feature-based software and systems product line engineering, or feature-based PLE, which has emerged as a proven and repeatable product line engineering and management (PLE) practice supported by commercial tool providers.

This document:

— provides the terms and definitions specific to feature-based PLE;

— defines how feature-based PLE is a specialization within the general ISO/IEC 26550 reference model for product line engineering and management;

— defines a reference model for the overall structure and processes of feature-based PLE and describes how the elements of the reference model fit together;

— defines interrelationships and methods for applying the elements and tools of the product line reference model;

— defines required and supporting tool capabilities.

In this document, products of feature-based PLE include digital work products that support the engineering of a system. Some of the artefacts are actually part of the delivered products, while other artefacts can be non-deliverable, such as physical or digital design models.

The intended audience for this document comprises:

— technology providers who wish to provide automated tool support for the reference model and processes described in this document;

— champions within an organization who wish to introduce feature-based PLE throughout that organization;

— IT staff within a PLE organization who will introduce and maintain the necessary technology to support feature-based PLE;

— practitioner stakeholders who will use the provided technology to practice feature-based PLE;

— technical and business managers who will sponsor and direct the methods necessary to practice feature-based PLE;

— university professors, researchers, corporate trainers, and other educators who will create and share pedagogical materials about feature-based PLE and its benefits.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

© ISO/IEC 2021 – All rights reserved

ISO/IEC/IEEE 12207, *Systems and software engineering — Software life cycle processes*

ISO/IEC/IEEE 15288, *Systems and software engineering — System life cycle processes*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC/IEEE 12207, ISO/IEC/IEEE 15288, and the following apply.

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

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

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

NOTE 1 For additional terms and definitions in the field of systems and software engineering, see ISO/IEC/IEEE 24765, which is published periodically as a "snapshot" of the SEVOCAB (Systems and software engineering – Vocabulary) database and is publicly accessible at www.computer.org/sevocab.

NOTE 2 Because feature-based PLE is a specialization of the more general product line engineering approach described in ISO/IEC 26550, some of the terminology used herein is noted as a specialization of the terminology from ISO/IEC 26550, with further details provided in Annex A.

3.1
bill-of-features
specification for a *member product* (3.8) in the *product line* (3.16), rendered in terms of the specific *features* (3.4) from the *feature catalogue* (3.5) that are chosen for that member product

3.2
bill-of-features portfolio
collection comprising the *bill-of-features* (3.1) for each *member product* (3.8) in a *product line* (3.16)

3.3
domain supersets
collection comprising the *feature catalogue* (3.5) and *shared asset supersets* (3.18)

3.4
feature
characteristic of a *member product* (3.8) in a *product line* (3.16) that distinguishes it from other member products in the product line

Note 1 to entry: Features can a) express the customer-visible or end-user-visible variability among the member products in a product line, or b) distinguish implementation variability not directly visible to a customer or end user except through non-functional differences such as price, performance, noise, weight, energy and more.

Note 2 to entry: In feature-based PLE, features express differences among member products. A capability or other characteristic common to all member products in the product line is not modelled as a feature.

Note 3 to entry: See Annex A for the definition of this term in ISO/IEC 26550.

3.5
feature catalogue
model of the collection of all the *feature* (3.4) options and *feature constraints* (3.6) available across the entire *product line* (3.16)

3.6
feature constraint
formal relationship between two or more *features* (3.4) that is necessarily satisfied for all *member products* (3.8)

© ISO/IEC 2021 – All rights reserved

3.7
feature language
syntax and semantics for the formal representation, structural taxonomy, and relationships among the concepts and constructs in the *feature catalogue* (3.5), *bill-of-features portfolio* (3.2), and *shared asset superset* (3.18) *variation points* (3.20)

3.8
member product
product belonging to the *product line* (3.16)

[SOURCE: ISO/IEC 26550:2015, 3.15, modified — The preferred term "application" has been removed.]

3.9
mutually exclusive
alternatives from which at most one is selected

3.10
mutually inclusive
alternatives from which zero or more are selected

3.11
PLE factory
technological, organizational, and business infrastructure and processes to support a *PLE factory configurator* (3.12) producing *product asset instances* (3.14) from *shared asset supersets* (3.18) based on a *bill-of-features* (3.1) for a *member product* (3.8)

3.12
PLE factory configurator
automated mechanism that produces assets for a specific *member product* (3.8) by processing the *bill-of-features* (3.1) for that member product, and exercising the *shared assets'* (3.17) *variation points* (3.20) in light of the *feature* (3.4) selections made in that bill-of-features

3.13
PLE factory development environment
toolset for creating, organizing, assembling, and maintaining a collection of elements in a *feature catalogue* (3.5), *bill-of-features portfolio* (3.2), *shared asset supersets* (3.18), and a hierarchy of a *product line* (3.16) of product lines

3.14
product asset instance
instantiation of a *shared asset* (3.17) specific to a *member product* (3.8), automatically produced by the *PLE factory configurator* (3.12), corresponding to a *bill-of-features* (3.1) for that member product

Note 1 to entry: A product asset instance is analogous to an application asset (ISO/IEC 26550) with the proviso that it is produced by the PLE factory configurator.

3.15
product instances
collection comprising the *bill-of-features portfolio* (3.2) and *product asset instances* (3.14)

3.16
product line
family of similar products with variations in *features* (3.4)

Note 1 to entry: See Annex A for the definition of this term in ISO/IEC 26550.

3.17
shared asset
software and systems engineering lifecycle digital artefacts that compose a part of a delivered *member product* (3.8) or support the engineering process to create and maintain a member product

Note 1 to entry: A shared asset is analogous to a domain asset (ISO/IEC 26550).

© ISO/IEC 2021 – All rights reserved

Note 2 to entry: Typical shared assets are requirements, design specifications or models for mechanical, electrical, and software, source code, build files or scripts, test plans and test cases, user documentation, repair manuals and installation guides, project budgets, schedules, and work plans, product calibration and configuration files, mechanical bills-of-materials, electrical circuit board and wiring harness designs, engineering management plans, engineering drawings, training plans and training materials, skill set requirements, manufacturing plans and instructions, and shipping manifests.

3.18
shared asset superset
representation of a *shared asset* (3.17) that includes all content needed by any of the *member products* (3.8)

3.19
variant
alternative that can be used to realize a particular *variation point* (3.20)

[SOURCE: ISO/IEC 26550:2015, 3.28, modified — the word "one" at the beginning of the definition has been removed; "may" has been changed to "can"; "particular variation points" has been changed to "a particular variation point"; note 1 to entry has been removed.]

3.20
variation point
identification of a specific piece of *shared asset superset* (3.18) content and a mapping from *feature* (3.4) selection(s) to the form of that content that appears in a *product asset instance* (3.14)

Note 1 to entry: In this document, all features express characteristics that differ among *member products* (3.8), which according to ISO/IEC 26550 would also make every feature a variation point. To avoid this redundancy, this document does not call out features as variation points.

Note 2 to entry: See Annex A for the definition of this term in ISO/IEC 26550. The definition in this document is more specific to feature-based PLE and the *PLE factory configurator* (3.12) approach of producing product asset instances from shared asset supersets.

4 Overview of feature-based product line engineering

4.1 General

This clause gives a brief informational overview of feature-based product line engineering, as a way of introducing key concepts that are important for understanding the purpose and content of this document.

Software and systems product line engineering emerged some time ago as way to engineer a portfolio of related products in an efficient manner, taking full advantage of the products' similarities while respecting and managing their differences. Here "engineer" means all of the activities involved in planning, producing, delivering, deploying, sustaining, and retiring products. Born in the software engineering field in the 1970s and 1980s and based largely on the concept of software reuse, PLE has long been known for delivering significant improvements in development time, cost, and quality of products in a product line[1].

Early attempts to capture and codify best practices recognized a dichotomy between two sides: product content that applies to multiple products and product content specific to a single product. Some referred to the two sides as domain engineering and application engineering, respectively, while others referred to core asset development vs. product development[1]. Application engineering was often said to include creating any content that happened to be used only in a single product, and promoting that content to domain status only if subsequently used in more. Application engineering included the obligation to choose and carry out a production strategy – that is, a way to turn the shared assets into products – that was potentially different for each type of shared asset, and was often manual and therefore labour-intensive and error-prone. Whatever they were called, the two sides stood on roughly equal footing in terms of the effort required to execute.

© ISO/IEC 2021 – All rights reserved