ISO/IEC 26552:2019 (E)

Software and systems engineering — Tools and methods for product line architecture design

Contents

Foreword

Introduction

1 Scope

2 Normative references

3 Terms and Definitions

4 Reference model for product line architecture design

4.1 Overview
4.2 Architecture management
4.3 Domain design
4.4 Asset management
4.5 Variability management in design
4.6 Application design

5 Architecture management

5.1 General
5.2 Architecture design planning
5.2.1 Principal constituents
5.2.1.1 Purpose
5.2.1.2 Inputs
5.2.1.3 Outcomes
5.2.1.4 Tasks
5.2.2 Establish architecture design goals
5.2.3 Define key procedures for architecture design
5.2.4 Define schedules and required resources for architecture design
5.2.5 Specify how to monitor, measure and control the effectiveness of architecture design
5.2.6 Document the architecture design plan
5.3 Architecture design enabling
5.3.1 Principal constituents
5.3.1.1 Purpose
5.3.1.2 Inputs
5.3.1.3 Outcomes
5.3.1.4 Tasks
5.3.2 Prepare for the architecture enablement
5.3.3 Develop and establish enabling capabilities and resources
5.3.4 Deploy capabilities and resources for architecture enablement
5.3.5 Improve architecture enablement capabilities and resources
5.4 Architecture design managing
5.4.1 Principal constituents
5.4.1.1 Purpose
5.4.1.2 Inputs
5.4.1.3 Outcomes
5.4.1.4 Tasks
5.4.2 Prepare for architecture management execution
5.4.3 Implement the architecture management plans
5.4.4 Close and prepare for the architecture management plan change

6 Domain design

6.1 General

6.2 Conceptual architecture design
6.2.1 Principal constituents
6.2.1.1 Purpose
6.2.1.2 Inputs
6.2.1.3 Outcomes
6.2.1.4 Tasks
6.2.2 Analyse problem space of the domain architecture
6.2.3 Synthesize potential solution alternatives
6.2.4 Formulate potential domain architecture(s)
6.2.5 Capture domain architecture concepts and properties
6.2.6 Hand off conceptualized domain architecture to users and other stakeholders
6.3 Domain architectural structure design
6.3.1 Principal constituents
6.3.1.1 Purpose
6.3.1.2 Inputs
6.3.1.3 Outcomes
6.3.1.4 Tasks
6.3.2 Develop architecture viewpoints for the product line
6.3.3 Develop models and views of the domain architecture
6.3.4 Relate the domain architecture to requirements
6.3.5 Relate the domain architecture to detailed design
6.4 Architectural texture design
6.4.1 Principal constituents
6.4.1.1 Purpose
6.4.1.2 Inputs
6.4.1.3 Outcomes
6.4.1.4 Tasks
6.4.2 Analyse common rules guiding realization
6.4.3 Define common ways to deal with variability at domain realization
6.4.4 Define common ways to deal with variability at application design and realization
6.4.5 Formulate architectural texture
6.5 Domain architecture documentation
6.5.1 Principal constituents
6.5.1.1 Purpose
6.5.1.2 Inputs
6.5.1.3 Outcomes
6.5.1.4 Tasks
6.5.2 Assess the domain architecture documentation for structure and texture
6.5.3 Hand off architecture documentation to downstream users
6.6 Domain architecture evaluation
6.6.1 Principal constituents
6.6.1.1 Purpose
6.6.1.2 Inputs
6.6.1.3 Outcomes
6.6.1.4 Tasks
6.6.2 Determine evaluation criteria for domain architecture
6.6.3 Establish measurement techniques for domain architecture
6.6.4 Review evaluation-related information for domain architecture
6.6.5 Analyse domain architecture and assess stakeholder satisfaction
6.6.6 Formulate findings and recommendations for domain architecture
6.6.7 Communicate evaluation results

7 Variability management in design

7.1 General
7.2 Internal variability in domain architecture
7.2.1 Principal constituents
7.2.1.1 Purpose
7.2.1.2 Inputs
7.2.1.3 Outcomes
7.2.1.4 Tasks
7.2.2 Identify newly added internal variability
7.2.3 Refine external variability into internal variability
7.2.4 Relate internal variability with variability in requirements
7.3 Variability model in architecture

7.3.1 Principal constituents
7.3.1.1 Purpose
7.3.1.2 Inputs
7.3.1.3 Outcomes
7.3.1.4 Tasks
7.3.2 Model variability in views of architecture(s)
7.3.3 Maintain variability model in architecture
7.3.4 Document variability in architecture
7.4 Variability mechanism in architecture
7.4.1 Principal constituents
7.4.1.1 Purpose
7.4.1.2 Inputs
7.4.1.3 Outcomes
7.4.1.4 Tasks
7.4.2 Identify variability mechanisms in architecture by category
7.4.3 Guide the use of variability mechanism category in architecture
7.4.4 Trace the usage status of variability mechanism category in architecture
7.4.5 Update variability mechanism category in architecture
7.5 Variability traceability in architecture
7.5.1 Principal constituents
7.5.1.1 Purpose
7.5.1.2 Inputs
7.5.1.3 Outcomes
7.5.1.4 Tasks
7.5.2 Define trace links among variability in different architectural artefacts
7.5.3 Define trace links between architectural artefacts and variability model

8 Asset management in design

8.1 General
8.2 Managing domain design artefacts as domain assets
8.2.1 Principal constituents
8.2.1.1 Purpose
8.2.1.2 Inputs
8.2.1.3 Outcomes
8.2.1.4 Tasks
8.2.2 Identify architectural artefacts managed as domain assets
8.2.3 Define configuration and annotation for domain architecture assets
8.3 Managing application design artefacts as application assets
8.3.1 Principal constituents
8.3.1.1 Purpose
8.3.1.2 Inputs
8.3.1.3 Outcomes
8.3.1.4 Tasks
8.3.2 Identify architectural artefacts managed as application assets
8.3.3 Define configuration and annotation for application architecture assets

9 Application design

9.1 General
9.2 Binding in architecture
9.2.1 Principal constituents
9.2.1.1 Purpose
9.2.1.2 Inputs
9.2.1.3 Outcomes
9.2.1.4 Tasks
9.2.2 Decide values of variabilities in architecture
9.2.3 Conduct bindings in architecture
9.2.4 Validate consistencies with bindings in requirements
9.2.5 Validate whether binding decisions adhere to the architectural texture
9.3 Application specific architectural structure design
9.3.1 Principal constituents
9.3.1.1 Purpose
9.3.1.2 Inputs
9.3.1.3 Outcomes
9.3.1.4 Tasks

9.3.2 Develop models of the application specific architecture
9.3.3 Validate whether the application specific architecture adheres to the architectural texture
9.3.4 Relate the application specific architecture to requirements
9.3.5 Relate the application specific architecture to detailed design
9.4 Application architecture documentation
9.4.1 Principal constituents
9.4.1.1 Purpose
9.4.1.2 Inputs
9.4.1.3 Outcomes
9.4.1.4 Tasks
9.4.2 Assess the application specific architecture documentation
9.4.3 Hand off application specific architecture documentation to downstream users
9.5 Application architecture evaluation
9.5.1 Principal constituents
9.5.1.1 Purpose
9.5.1.2 Inputs
9.5.1.3 Outcomes
9.5.1.4 Tasks
9.5.2 Determine application specific evaluation criteria
9.5.3 Establish application specific measurement techniques
9.5.4 Review evaluation-related information for application architecture
9.5.5 Analyse application architecture and assess stakeholder satisfaction
9.5.6 Formulate findings and recommendations for application architecture
9.5.7 Communicate evaluation results with application specific stakeholders

Annex A (informative) Cross-reference with ISO/IEC/IEEE 42020, ISO/IEC/IEEE 42010 and ISO/IEC/IEEE 15288

Annex B (informative) Variability specification elements in ADL

Annex C (informative) Architecture structure and texture example

Page count: 61