

Vworks and HVP

Generic Software Development Plan

**PROPRIETARY**

|  |  |  |
| --- | --- | --- |
| Prepared by: |  | VxWorks and HVP Engineering Manager |
| Checked by: | Shree Vidya Jayaraman | SQA Lead |
| Approved by: | Martin Cote (HVP)  Kitty Kong (VxWorks) | Director, Engineering |

|  |  |
| --- | --- |
| Document ID: | WRGeneric\_SDP |
| Author: |  |
| Version: |  |
| Version Date: |  |
| Status: |  |

**Copyright Notice**

Copyright © 2020 Wind River Systems, Inc.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means without the prior written permission of Wind River Systems, Inc.

Wind River, Simics, Tornado, and VxWorks are registered trademarks of Wind River Systems, Inc. Helix, Pulsar, Rocket, Titanium Cloud, Titanium Control, Titanium Core, Titanium Edge, Titanium Edge SX, Titanium Server, and the Wind River logo are trademarks of Wind River Systems, Inc. Any third-party trademarks referenced are the property of their respective owners. For further information regarding Wind River trademarks, please see:

[www.windriver.com/company/terms/trademark.html](http://www.windriver.com/company/terms/trademark.html)

This product may include software licensed to Wind River by third parties. Relevant notices (if any) are provided for your product on the Wind River download and installation portal, Wind Share:

<http://windshare.windriver.com>

Wind River may refer to third-party documentation by listing publications or providing links to third-party websites for informational purposes. Wind River accepts no responsibility for the information provided in such third-party documentation.

**Corporate Headquarters**

Wind River

500 Wind River Way

Alameda, CA 94501-1153

U.S.A.

Toll free (U.S.A.): +1-800-545-WIND

Telephone: +1-510-748-4100

Facsimile: +1-510-749-2010

For additional contact information, see the Wind River website:

[www.windriver.com](http://www.windriver.com)

For information on how to contact Customer Support, see:

[www.windriver.com/support](http://www.windriver.com/support)

| Revision History | | | |
| --- | --- | --- | --- |
| Date | Version | By | Description of Change |
| 07/21/2020 | 0.1 |  | Intial Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[1 Introduction 6](#_Toc46227312)

[1.1 Scope 6](#_Toc46227313)

[1.2 Applicable Documents 6](#_Toc46227314)

[1.3 Relationship to Other Planning Documents 7](#_Toc46227315)

[1.4 Glossary and Acronyms 8](#_Toc46227316)

[2 Personnel and Responsibilities 9](#_Toc46227317)

[2.1 Roles 9](#_Toc46227318)

[2.2 Project Organization 9](#_Toc46227319)

[3 Standards – Check with Kitty/Martin 10](#_Toc46227320)

[3.1 Software Requirements Standard 10](#_Toc46227321)

[3.2 Software Design Standard 10](#_Toc46227322)

[3.3 Coding Standard 10](#_Toc46227323)

[4 Lifecycle Activities 11](#_Toc46227324)

[4.1 Overview 11](#_Toc46227325)

[4.2 Execution Strategy 11](#_Toc46227326)

[4.3 Planning 11](#_Toc46227327)

[4.3.1 Goals 11](#_Toc46227328)

[4.3.2 Inputs and Start Criteria 11](#_Toc46227329)

[4.3.3 Overview of Activities 11](#_Toc46227330)

[4.3.4 Outputs and End Criteria of Software Planning Phase 13](#_Toc46227331)

[4.4 Software Requirements Analysis 13](#_Toc46227332)

[4.4.1 Goals 13](#_Toc46227333)

[4.4.2 Inputs and Start Criteria 13](#_Toc46227334)

[4.4.3 Overview of Activities 14](#_Toc46227335)

[4.4.4 Outputs and End Criteria of Software Requirements Analysis 14](#_Toc46227336)

[4.5 Plan for Verification Testing & Validation Strategy 15](#_Toc46227337)

[4.5.1 Goals 15](#_Toc46227338)

[4.5.2 Inputs and Start Criteria 15](#_Toc46227339)

[4.5.3 Overview of Activities 15](#_Toc46227340)

[4.5.4 Outputs and End Criteria of Validation Phase 16](#_Toc46227341)

[4.6 Software Design and Development 16](#_Toc46227342)

[4.6.1 Software Architecture Design Phase 16](#_Toc46227343)

[4.6.2 Detailed Design and Development Phase 17](#_Toc46227344)

[4.6.3 Code Integration Phase 18](#_Toc46227345)

[4.7 Software Testing (Verification) 19](#_Toc46227346)

[4.7.1 Goals 19](#_Toc46227347)

[4.7.2 Inputs and Start Criteria 20](#_Toc46227348)

[4.7.3 Overview of Activities 20](#_Toc46227349)

[4.7.4 Outputs and End Criteria of Software Testing 20](#_Toc46227350)

[4.8 Validation 21](#_Toc46227351)

[4.8.1 Goals 21](#_Toc46227352)

[4.8.2 Inputs and Start Criteria 21](#_Toc46227353)

[4.8.3 Overview of Activities 21](#_Toc46227354)

[4.8.4 Outputs and End Criteria of Validation 22](#_Toc46227355)

[5 Defects and Change Management 23](#_Toc46227356)

[6 Software Development Environment 24](#_Toc46227357)

[7 Plan Review and Approval 25](#_Toc46227358)

**List of Tables**

[Table 1: Applicable Documents 7](#_Toc33169002)

# Introduction

## Scope

This plan divides the selected software lifecycle for the development of VxWorks and HVP into elementary activities. It describes in detail for each phase the following:

* Goals
* Scope
* Inputs
* Outputs

This document is created during the program release planning, and once approved (baselined in the Configuration Management (CM) system) is changed by filing change requests, and following the change management process.

## Applicable Documents

The table below lists all documents referenced within this .

|  |  |
| --- | --- |
| Ref. | Title |
|  | *Wind River VxWorks and HVP Process Defintion* |
|  | *Wind River VxWorks and HVP Generic Glossary* |
|  | *Wind River VxWorks and HVP Generic Requirements Specifications Standard* |
|  | *Wind River VxWorks and HVP Generic Design Specifications Standard* |
|  | *Wind River VxWorks and HVP Generic Coding Standard* |
|  | *Wind River VxWorks and HVP Generic Software Quality Assurance Plan* |
|  | *Wind River VxWorks and HVP Generic Software Configuration Management Plan* |
|  | *Wind River VxWorks and HVP Generic Overall Product Test Strategy* |
|  | *Wind River VxWorks and HVP Generic Validation Plan* |
|  | *Wind River Product Release Profile (VxWorks and HVP separately)* |
|  | *Wind River VxWorks and HVP Generic Roles and Responsibilities* |
|  | *Wind River VxWorks and HVP Generic Tools and Environments* |
|  | *Wind River VxWorks and HVP Generic Training Plan* |
|  | *Wind River VxWorks and HVP Verification/Test Plan Documents* |

Table : Applicable Documents

## Relationship to Other Planning Documents

The software development lifecycle activities for the Wind River Linux project are described in the following documents:

* Wind River VxWorks and HVP Process Definition [Ref.1]
* Wind River VxWorks and HVP Generic Software Quality Assurance Plan [Ref.6]
* Wind River VxWorks and HVP Generic Software Configuration Management Plan [Ref.7]
* Wind River VxWorks and HVP Generic Overall Product Test Strategy [Ref.8]
* Wind River VxWorks and HVP Generic Validation Plan [Ref.9]
* Wind River VxWorks and HVP Generic Product Release Profile [Ref.10]
* Wind River VxWorks and HVP Generic Tools and Environments [Ref.12]

The process definition facilitate the activities performed throughout the development lifecycle and by the support functions such as Program Release Management, Configuration Management, and SQA.

The Software Quality Assurance Plan (SQAP) specifies the activities for quality assurance during the lifecycle of the program.

The Software Configuration Management Plan (SCMP) describes the processes that will be used to meet configuration management objectives of the program release.

The Overall Product Test Strategy (PTS) describes the program level test strategy, including the overall goals and activities of the product testing – including the interaction, coverage and integration of the team-specific development and test documents. ?? check with Kitty

The Feature Test Plan & Release Test Strategy describes the verification testing activities that will be used to meet the testing requirements of the program release.

The Validation Plan describes the validation activities for the program.

The Product Release Profile establishes an overall plan for release management. It identifies the overall tasks and engineering management planning required to control the design, development, and tests associated with the program.

The Tools and Environments document lists the tools and environments for the product.

## Glossary and Acronyms

For a list of terms and acronyms used in this document, refer to the Glossary document [Ref.2].

# Personnel and Responsibilities

## Roles

Roles and responsibilities are defined in the Wind River Roles and Responsibilities document [Ref.14]. Independence of roles is also defined in the same document.

## Project Organization

The project organization is described in the Product Release Profile / Plan document [Ref.10].

# Standards – Check with Kitty/Martin

## Software Requirements Standard

The Wind River Generic Requirements Specification Standard [Ref.3] defines the categories, conventions, and notation to be used for all software requirements. It defines a set of rules to be followed when writing software requirements, to ensure each requirement is unique, clear, verifiable, and traceable back to marketing or customer requirements.

## Software Design Standard

The Wind River Generic Design Specification Standard [Ref.4] defines the methods, rules, and tools to be used to develop the software architecture and design requirements for security. It defines a set of rules to be followed when writing designs, to ensure each design complies with the software requirements, is clear, modular and detailed, and complies with all design objectives. The document specifies the level of detail that is required, along with how to capture information flows or activity diagrams.

## Coding Standard

The Wind River Generic Coding Standard [Ref.5] of the product defines the rules that must be followed for all code and accompanying documentation included in the source code. The conventions encourage higher quality code and uniformity in the code.

# Lifecycle Activities

## Overview

Wind River VxWorks and HVP programs follow the corporate Product Development Lifecycle (PDLC), Agile Methodology and VxWorks and HVP Process Defintion [Ref.1],

The following sections identify the intended approach to software development based on the standard process defintion. Any deviations from the VxWorks and HVP Process Defintion [Ref.1] will be explicitly called out as a note.

## Execution Strategy

The programs follow the full Agile Methodology and Continuous Integration??

## Planning

This section describes the framework for the planning of the product release, including goals, start and end criteria, main activities, and work products. Release Planning applies to all phases and sets the framework for successful software development.

### Goals

Release Planning shall define the strategy for software development, integration, verification and validation. (Verification and validation planning are addressed in Section 4.5.)

Software Planning shall specify an adequate set of methods, tools, and techniques used for all phases of the life cycle.

### Inputs and Start Criteria

Inputs:

* Product Backlog – market/product requirements as Epics
* Charter
* Overall release schedule

Start criterion for Software Planning is:

* Engineering Program Manager (EPM) assigned to product release management

### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Preparation of Product Release Profile [Ref.10]
* Prepare Software Development Plan (this document)
* Prepare Software Configuration Management Plan [Ref.7]
* Prepare Tools and Environments document [Ref.12]
* Prepare Training Plan [Ref.13]
* Prepare Overall Product Test Strategy [Ref.8]
* Appoint all personnel involved in all phases of the lifecycle, including management activities, with appropriate training, experience, and qualification in accordance with the key competencies described in the Roles and Responsibilities document [Ref.11].
* Tool access privileges are documented in the Software Configuration Management Plan
* Communicate responsibilities to all persons, departments and organizations responsible for carrying out activities in the applicable software lifecycle phases.
* Assessment of the adequacy of the Process Definitions [Ref.1]
* Review of high-level Epics and requirements

Notes: The plan documents are created generic at the product level applicable to all VxWorks and HVP product releases.

Quality Activities:

* Prepare Software Quality Assurance Plan [Ref.6]
* Conduct Quality Audit
* Verify Plans, at a minimum:
  + Review of Generic Software Development Plan
  + Review of Generic Software Quality Assurance Plan
  + Review of Generic Software Configuration Management Plan
  + Review of Product Profile
  + Review of Generic Training Plan
  + Review of Overall Product Test Strategy
* Ensure that quality assurance activities have been performed as planned in the Software Quality Assurance Plan [Ref.6].

### Outputs and End Criteria of Software Planning Phase

Outputs are:

* Generic Software Quality Assurance Plan [Ref.6]
* Product Profile [Ref.10]
* Genric Software Configuration Management Plan [Ref.7]
* Generic Software Development Plan (this document)
* Generic Tools and Environments document [Ref.12]
* Generic Training Plan [Ref.13]
* Overall Product Test Strategy [Ref.8]
* Updated SQA Report (Audit/Verification/Review)

The end criteria of Software Planning are:

* Required generic planning documents are under configuration management (CM)

## Software Requirements Analysis

### Goals

The objective of Software Requirements Analysis is to identify, analyze and define the requirements for the software in terms of functionality.

### Inputs and Start Criteria

Inputs:

* High level Epics and requirements
* Requirements Specification Standard [Ref.3]
* Generic Software Configuration Management Plan [Ref.7]
* Product Profile [Ref.10]
* Generic Software Quality Assurance Plan (for verification activities) [Ref.6]

Start criteria for Software Requirements Analysis are:

* The high level Epics and requirements are entered into the requirements management system.
* Approved Generic Software Configuration Management Plan
* Approved Requirements Specifications Standard

### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Derive release specific requirements from the high-level Epics and requirements
* Enter the release specific requirements as Epic user stories in the Requirements Management system. Document release specific requirements according to the Requirements Specifications Standard ?? check with kitty and Martin
* Create the baseline that constitutes the committed set of requirements for the respective release cycle.

Quality Activities

* Verify that activities have been performed as planned
* Ensure Forward Traceability from high-level Epics and requirements to release specific requirements
* Ensure Backward Traceability of release specific requirements to high-level Epics and requirements
* Verify that release specific requirements are specified in the Verification (testing) plan documents [Ref.14] and Validation Plan [Ref.10].
* Ensure that assurance activities have been performed as planned in the Software Quality Assurance Plan [Ref.6].

### Outputs and End Criteria of Software Requirements Analysis

The outputs shall be:

* Committed set of requiremnets for the release
* Updated SQA Report (Audit/Verification/Review)

The end criteria of Software Requirements Analysis shall be:

* High-level Epics and requirements are covered by release specific requirements.
* Committed set of requiremnets for the release in the Requirement Management system .
* Committed set of requirements for the release approved for implementatiuon by the Key Stakeholders.— Product Architect in collaboration with the Scrum Team.
* Committed set of requirements is baselined and under CM.

## Plan for Verification Testing & Validation Strategy

### Goals

The objective is to develop a plan for the Verification Testing and Validation strategy of the committed set of release requirements.

### Inputs and Start Criteria

Inputs:

* High-level Epics and requirements
* Committed set of requirements for the release

Start criteria for Verification and Validation phase are:

* Committed set of requirements for the release approved for implementatiuon by the Key Stakeholders.— Product Architect in collaboration with the Scrum Team.

### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Specify verification (feature and release testing) and validation strategy
* Specify Test Environment and update the Generic Tools and Environment document [Ref.15], if necessary.
* Prepare the Feature Test (Verification) Plan and Release Test Strategy [Ref.14]
* Prepare the Validation Plan [Ref.10] ???

Quality Activities:

* Ensure that assurance activities have been performed as planned in Generic Software Quality Assurance Plan [Ref.6]
* Ensure Forward Traceability from committed set of release requirement to the verification tests and validation strategy specifications.
* Ensure Backward Traceability from Software Verification tests and Validation strategy specifications to the committed set of release requirements.

### Outputs and End Criteria of Validation Phase

The Outputs shall be:

* Verification (Testing) plan/strategy documents [Ref.14]
* Validation Plan [Ref.10] ??
* Updated Generic Tools and Environments document [Ref.12], if applicable
* Updated SQA Report (Audit/Verification/Review)

The end criteria of Verification and Validation for software shall be:

* All committed release requirements are covered by Verification tests and Validation strategy
* Key Stakeholders approve the (Verification) Plan & Strategy docuements and Validation Plan
* All work products, including (Verification) Plan & Strategy docuements and theValidation Plan, are baselined and under CM

## Software Design and Development

### Software Architecture Design Phase

#### Goals

The objective of the Software Architecture Design is to create high level design (HLD) document for each requirement, as applicable.

#### Inputs and Start Criteria

Inputs:

* Requirements are entered as Epics in the Requirements Management system
* Design Specification Standard [Ref.4]

Start criteria for Software Architecture Design are:

* High level Epics and requirements are available

#### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Create design documents (HLDs)
  + Identify Epics that requires a HLD to be created
  + Create HLD, conforming to the Design Specification Standard

Quality Activities:

* Verify that activities have been performed as planned (Green light review)
* Ensure Forward Traceability from high-level Epics and requirements to their HLDs
* Ensure Backward Traceability of the HLDs to high-level Epics and requirements
* Ensure that assruance activities have been performed as planned in the Software Quality Assurance Plan [Ref.6]

#### Outputs and End Criteria of Software Design

The outputs shall be:

* HLDs created for each requirement as applicable
* Traceability between HLD and requirements is recorded into the Epic and HLD document
* Updated SQA Report (Audit/Verification/Review)

The end criteria of the Software Design shall be:

* Key Stakeholders review and approve the Software requirements and their HLDs
* Approved HLDs are baselined and under CM

### Detailed Design and Development Phase

Product code is developed using the Coding Standard [Ref.5].

#### Inputs and Start Criteria

Inputs:

* Requirements entered Epics in the Requirements Management system
* Coding Standard
* HLDs available

Start criteria for Software Architecture Design are:

* Epics (requirements) are available

#### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Develop code
* Perform developer tests
* Perform peer review of code (developed code and performed developer test results)

Quality Activities

* Verify that activities have been performed as planned
* Ensure that assurance activities have been performed as planned in the Software Quality Assurance Plan [Ref.6]

#### Outputs and End Criteria of Software Design and Development

The outputs shall be:

* Code, Unit tests (Developer tests)
* Code review reports
* Updated SQA Report (Audit/Verification/Review)
* Updated or newly created HLD, as appropriate

The end criteria of Software Design and Development shall be:

* Code has been reviewed according to the Peer Review Process defined [Ref.1]
* Work products are baselined and under CM.

### Code Integration Phase

Code is integrated into a single cohesive product.

#### Goals

The objective of Code Integration shall be to implement the product requirements.

#### Is and Start Criteria

Inputs:

* Developed code, patches
* Documentation ready for submission, if applicable

Start Criteria for Code Integration:

* Code that has passed the Peer Review Process

#### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Integrate Software Code
  + Merge developed code
  + Compile code, unit test, assess that code meets requirements

Quality Activities:

* Verify that activities have been performed as planned
* Ensure that assurance activities have been performed as planned

#### Outputs and End Criteria of Coding

The outputs shall be:

* Source Code
* Updated SQA Report (Audit/Verification/Review)

The end criteria of Coding shall be:

* Requirements have been implemented and integrated
* Source Code is verified
* Work products are baselined and under CM

## Software Testing (Verification)

The Software Testing will verify that the requirements of the Wind River Linux Software Design Specification Standard [Ref.4] have been met.

### Goals

Testing shall verify that the requirements of the design have been met. It shall show that the software performs its intended function.

### Inputs and Start Criteria

Inputs:

* Source code
* Test specification as documented in the Verification (Testing) Plan and Strategy documents [Ref.14].

Start criteria for Software Testing shall be:

* Software design and development phase completed
* Test specification approved

### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Execute tests as specified in the Verification (Testing) Plan and Strategy documents [Ref.14].
* Update Verification (Testing) Plan and Strategy documents [Ref.14], if applicable
* Update Verification Test Report

Quality Activities:

* Verify that activities have been performed as planned
* Review and Verification (Testing) Plan and Strategy documents [Ref.14]
* Review and verify Verification Test Report
* Ensure that assurance activities have been performed as planned in the Software Quality Assurance Plan [Ref.6].

### Outputs and End Criteria of Software Testing

The outputs shall be:

* Updated Verification (Testing) Plan and Strategy documents [Ref.14]
* Updated Verification Test Report
* Updated SQA Report (Audit/Verification/Review)
* Test results

The end criteria of tests shall be:

* All Tests executed
* Key Stakeholders review and approve the Test results as acceptable
* Updated Verification (Testing) Plan and Strategy documents [Ref.14] approved
* Updated Verification Test Report approved
* Work products have been baselined and under CM

For further details, see also Wind River Linux Software Test (Verification) Plan/Test Strategy.

## Validation

The software validation of the Wind River Linux software shall ensure compliance with high-level Epics and requirements.

### Goals

Ensure that an independent validator performs the activity of demonstration by analysis and test and that the Wind River Linux software meets, in all respects, its specified high-level Epics and requirements, including customer requirements.

### Inputs and Start Criteria

Inputs:

* High-level Epics and requirements
* Wind River Linux Validation Plan [Ref.10]

Start criteria for Validation shall be:

* Sprint is complete and Epics/User Story (ies) is accepted and ready for validation.

### Overview of Activities

The following activities shall be performed:

Technical Activities:

* Execute validation strategy as specified in the Wind River Linux Validation Plan [Ref.10]

Note: Validation can be initiated during the Software Testing (Verification).

* Prepare Wind River Linux Validation Report

Quality Activities:

* Verify that activities have been performed as planned
* Review and verify Wind River Linux Validation Plan
* Review and verify Wind River Linux Validation Report
* Ensure that Validation activities have been performed as planned

### Outputs and End Criteria of Validation

The outputs shall be:

* Updated Wind River Linux Validation Plan [Ref.10] - if applicable.
* Wind River Linux Validation Report
* Updated SQA Report (Audit/Verification/Review)

The end criteria of Validation shall be:

* Validation has been completed and recorded
* Review of validation results have been conducted
* Work products have been baselined and under CM

# Defects and Change Management

Defects (A.K.A bugs) may be filed at any time throughout the software development lifecycle, both prior to and after the release of the software. Change requests are always used to track changes to approved configuration items (which have been reviewed and put under configuration management).

The defect management and change management process is documented in the Process Definition [Ref.1].

# Software Development Environment

Project staff members all have access to the project development environment, including a personal computer with networked access to build servers, embedded target computers, simulators and software development tools as required to do their job. Wind River is a multi-national company, so team members may or may not be co-located. In the case where co-location is not possible, teams communicate via email, instant messenger, WebEx and/or telephone.

The tool environment to be used for this project will be defined in the Generic Tools and Environments document [Ref.12] and Product Release Profile [Ref.10].

The Linux QMS [Ref.3] and the individual plan documents referenced above in section 1.2 describe the processes and tools to be used throughout the software development lifecycle.

# Plan Review and Approval

This document shall be modified and controlled as specified in the Wind River Linux Software Configuration Management Plan [Ref.7].

The document shall be modified and controlled and any recommended changes to this document shall be reviewed and approved prior to every program release.  The Engineering Director/Manager shall make the necessary updates and submit the plan for internal review (with key stakeholders) and approval.