Stakeholder Requirement to Final SW for Product

Software Product Development Workflow

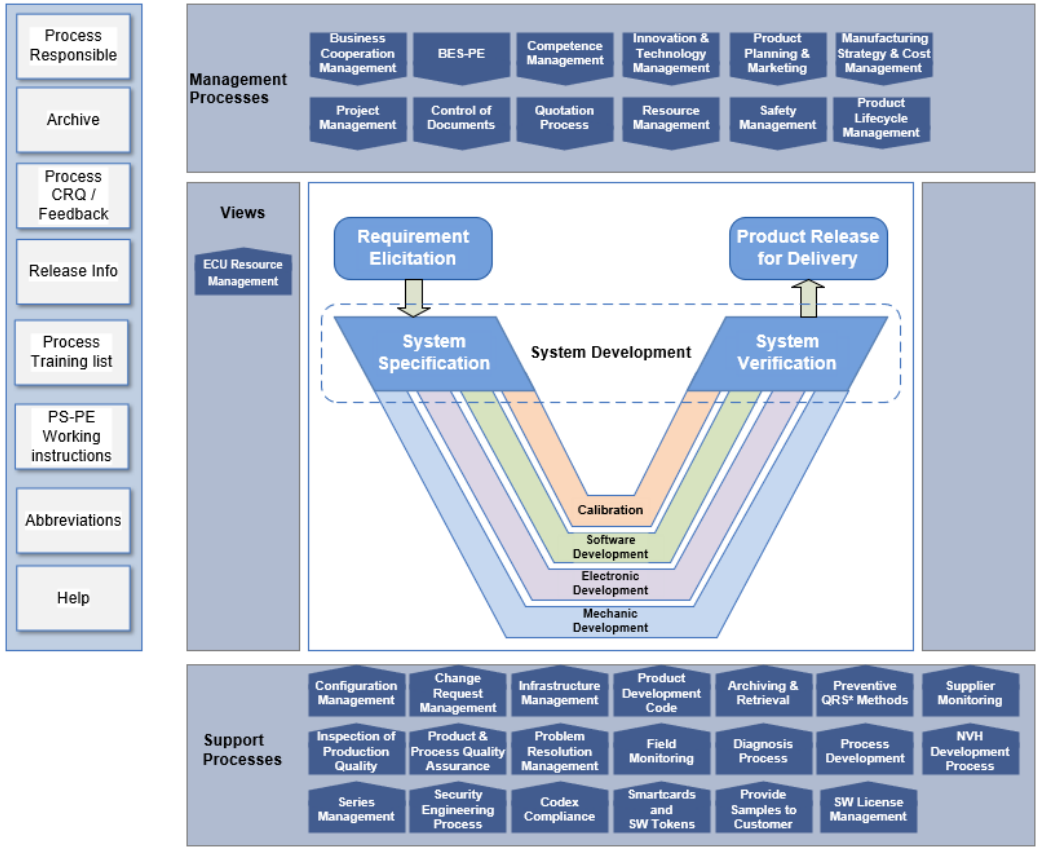
BOSCH

Agenda

- Introduction – Software Product Development Lifecycle
- ASPICE Mapping to PS-PE Product Development
- Software Product Development Workflow
  - Pre-Software Processes
  - Software Requirement Engineering
  - Software Architecture Design
  - Software Detailed Design and Implementation
  - Software Unit Verification
  - Software Integration and Integration Test
  - Software Qualification test
  - Program version and Initial Calibration Data Delivery
  - Software Release Meeting
  - Final Calibration and Container generation
  - Post Software Development Processes

Software Product Development Workflow

Software Development in EM-PE Product Development

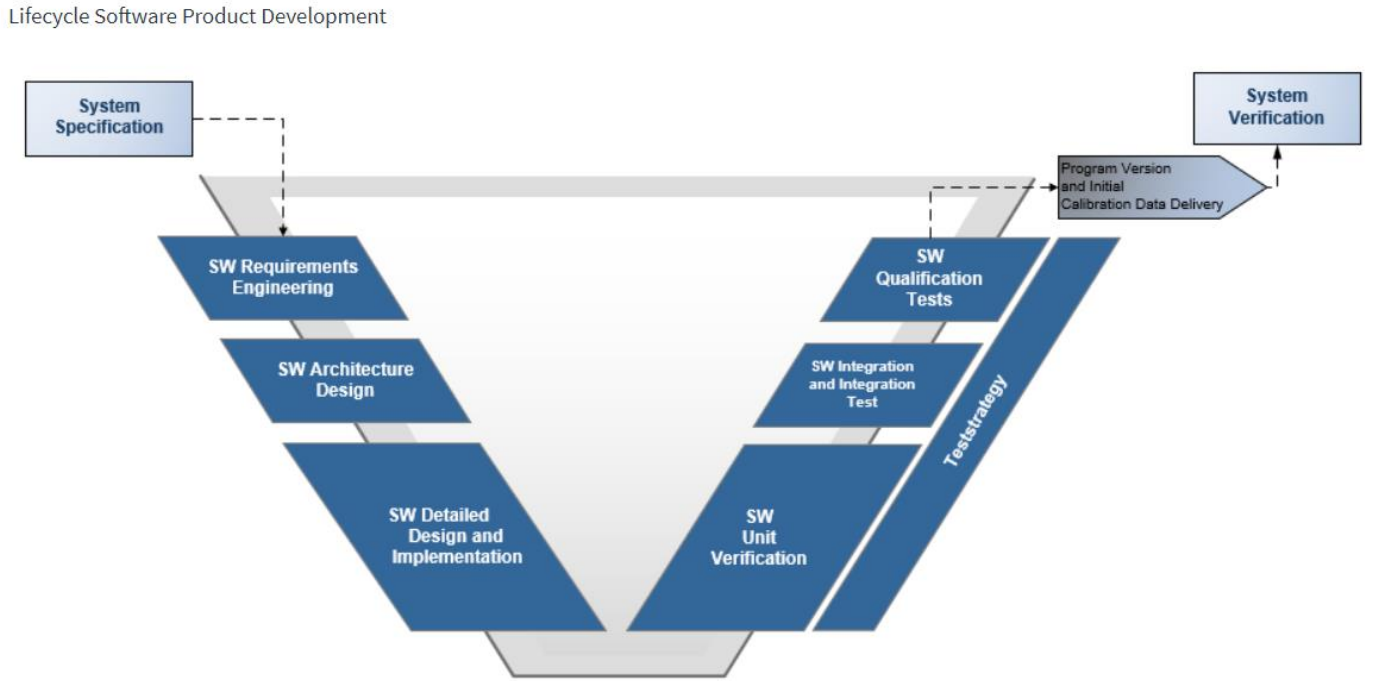


BOSCH

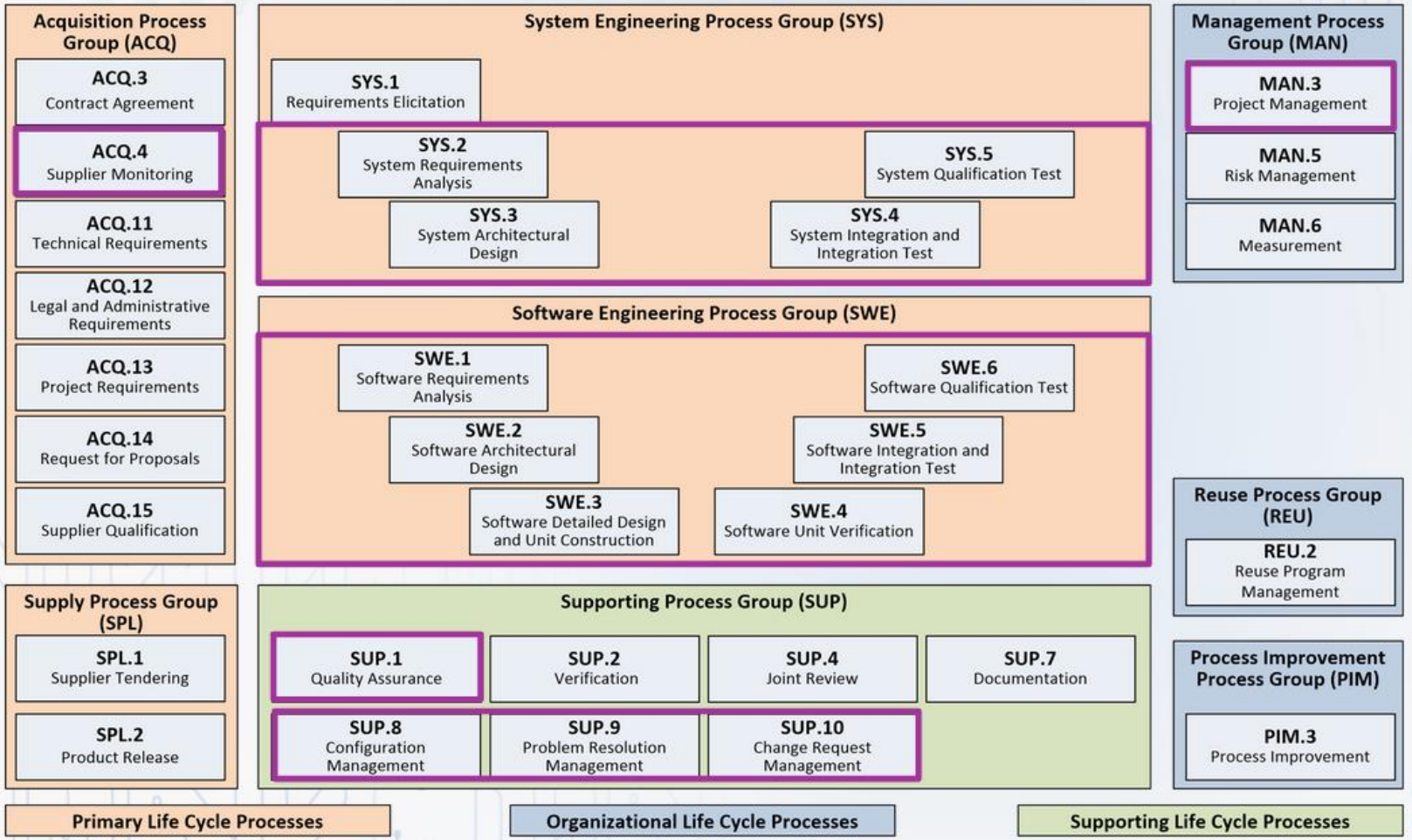
Software Product Development Workflow

Introduction – Software Product Development Lifecycle

- **SW Product Development describes the development lifecycle of a single program version from requirements analysis till delivery.**



Process Reference Model (PRM)* – Processes and Process Groups



Processes are selected based on relevance for the organization

Processes according to Automotive SPICE Guidelines

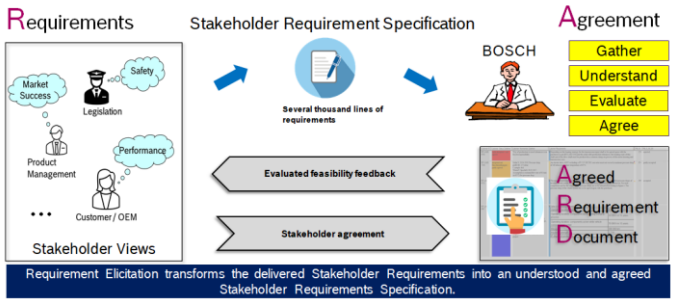
*Automotive SPICE v3.1

BOSCH

Software Product Development Workflow

Requirement Elicitation

- The Requirements Elicitation process describes how to gather, obtain, understand, evaluate and agree external and internal Stakeholder Requirements Specifications.
- The process transforms the delivered Stakeholder Requirements Specification into the agreed and baselined Stakeholder Requirements Specification. If needed, a deviation list is created.



System Requirement Analysis

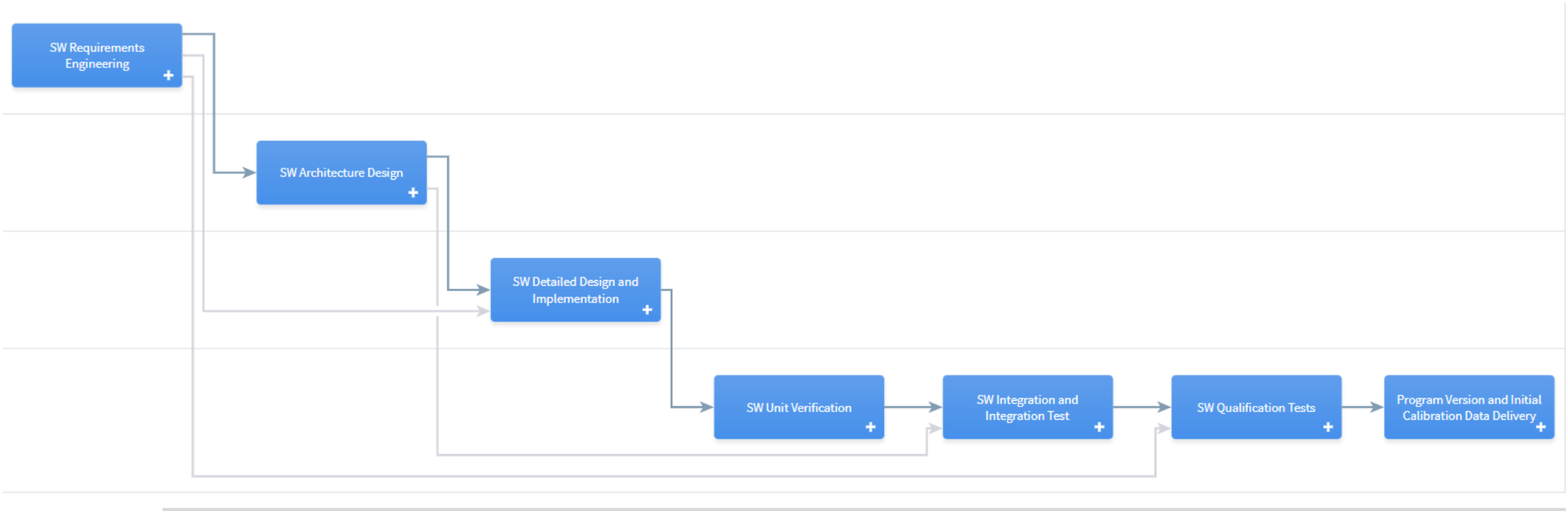
- Prepare Development Environment
- Identify system boundaries, interfaces, and environment
- Identify system requirements
- Analyze dependencies of the system requirements
- Analyze technical feasibility and define verification criteria
- Review system requirements with Domain Responsible
- Release System and Domain (SW, HW, CAL etc) requirement specification

System Architecture Design

- Prepare System architecture environment
- Derive Project System Architecture
- Design System User Function/Chain
- Design System Element
- Communicate and Publish System Architecture

Software Product Development Workflow

Process Flow of Software Development



Software Product Development Workflow

Software Requirement Engineering –

SW Requirement Engineer / Previously it was SWSD

This process describes how the software related parts of the System Requirements Specification and the SysArchSpec: System Architecture Specification (including interface definitions) are transformed into a SW Requirements Specification.

1. Specify the SW requirements
2. Analyze the SW requirements
3. Structure the SW requirements
4. Develop verification criteria
5. Establish bidirectional traceability of SW requirements

SW Requirement Freeze

Software Architecture Design

SW Architect

The purpose of this workflow is to provide a SW Architectural Design down to the level of SW components (SWCs). This mainly includes:

- to define the SW structure (that means to determine the SWCs),
- to allocate the software requirements to the SWCs,
- to define the interfaces between all SWCs and the scheduling of SWCs,
- to define criteria how the SW Architectural Design can be verified,
- and to define central/global elements and directives (e.g. SW coding guidelines).

The SW Architecture Design supports an early definition of SWCs, their interfaces/scheduling and global SW elements used by more than one SWC. This avoids iterations during SW development.

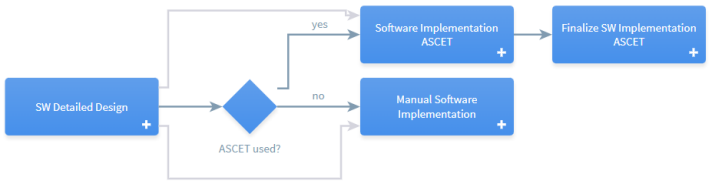
SW Design Freeze

BOSCH

Software Product Development Workflow

Software Detailed Design and Implementation – Software Dev Engineer

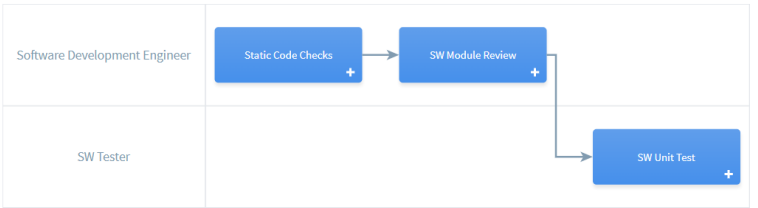
This process describes the activities SW detailed design and SW unit implementation inside a project.
The detailed design contains the detailed concept for a SW component and its units including why the design was chosen.
SW components and their units are implemented according to this design.



Software Unit Verification – Software Tester

The main goal/purpose of the SW Unit Verification is to cover the following points:

1. Static Code Checks - (Static Unit Verification)
2. SW Module Review - (Static Unit Verification)
3. Verification that the code corresponds to Software Module Specification (swMS) in SW Unit Test Case Specification - (Dynamic Unit Verification)
4. Finding errors in code (inaccessible code, infinite loops, division by zero, overruns, data types...) - (Dynamic Unit Verification)
5. Reaching a Coverage (C1) - (Dynamic Unit Verification)



SW Implementation Freeze

Software Product Development Workflow

Software Integration and Integration Test –

SW Integrator

This process describes the integration of the Software Components into larger software items up to a complete integrated software and verification of the integrated software against software architectural design.

SW Integration

- Set the system constants
- Create customer-specific hardware configurations
- Collect EEPROM configuration information
- Configure EEPROM layout
- Configure the process sequence
- Follow integration strategy to create a program version
- Verify the program version created/stored in the CM system

SW Integration Test

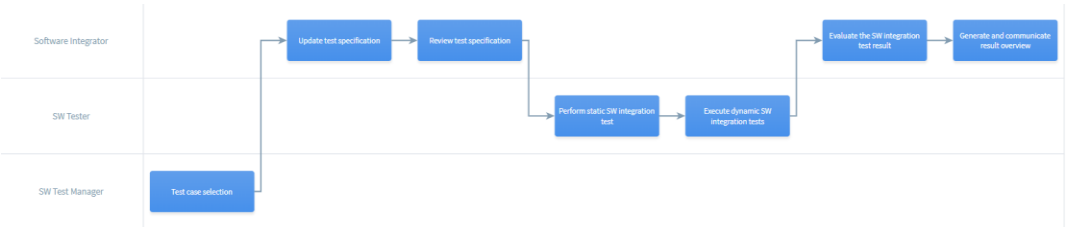
- The SW integration test verifies the correct integration of each component against the software architecture into an integrated SW (PVER). The test process is organized in a two-level test approach including a static and dynamic test scenario.

SW Integration Freeze

Software Integration Tests
(previously checklist)

The Integration tests performed are:

- RTE Warning Analysis
- Port Configuration Check
- Scheduling Check
- Plan Checker
- System Constant Setting check
- Dynamic Interface Test
- Interface check
- Runtime Resource Measurement
- Memory Resources
- Short test

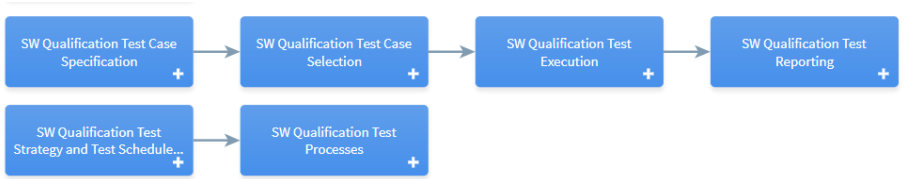


Software Product Development Workflow

Software Qualification test –

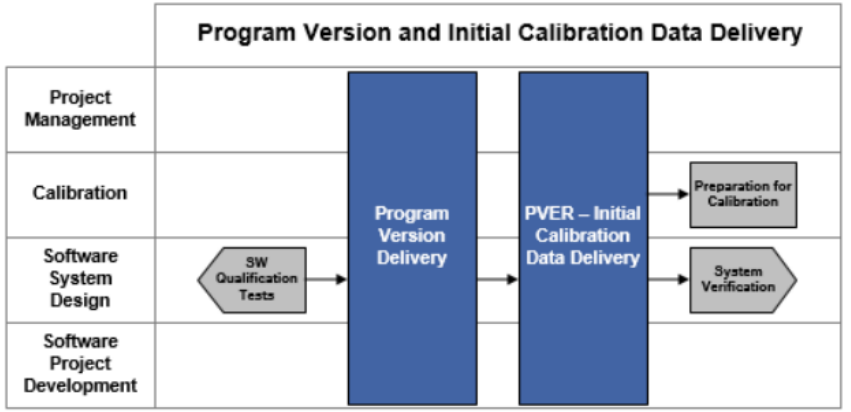
Testing team

- This workflow describes SW Qualification Test of the integrated software against accepted SW Requirements.
- Human resources are SW Test Manager, SW Test Designer and SW Tester. Technical resources are Test Environment HIL (hardware in the loop).
  Resources are provided with estimation model and agreed with Subproject Manager SW



Program version and Initial Calibration Data Delivery

This process describes the procedure for reviewing and delivering an ECU program version



SW Delivery Open Software

Software Product Development Workflow

Software Release Meeting

Motivation

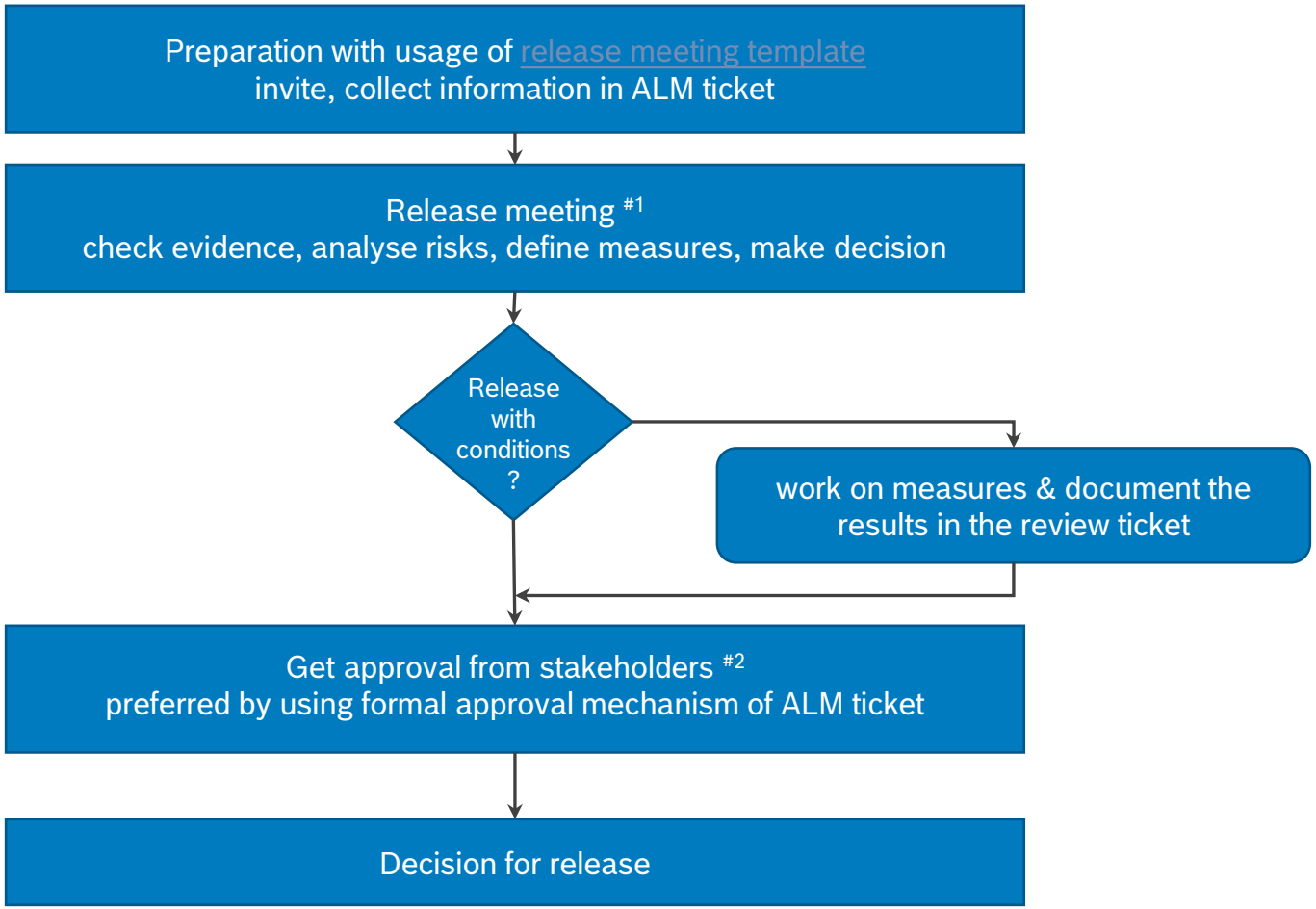
- check for completeness of release
- communication between all stakeholders, common understanding
- risk assessment and problem resolution
- traceability of released software for distribution

Stakeholder

- line manager of affected domains
- project manager, safety responsibles and quality representatives
- additional participants based on project situation and special topics

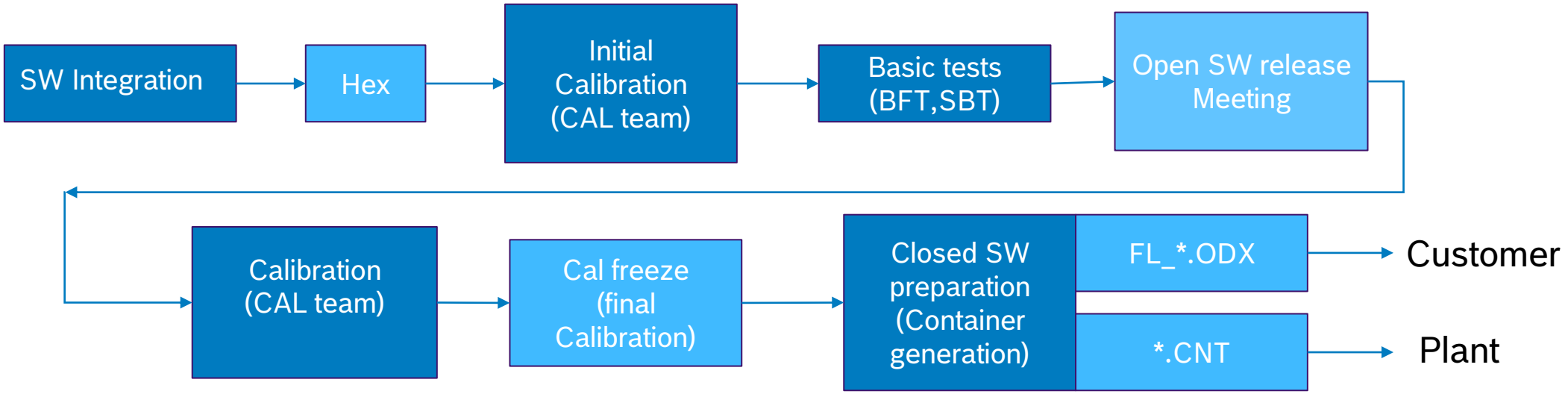
Relevance

- A release meeting prior to the delivery is mandatory for software versions that are handed over to the customer



BOSCH

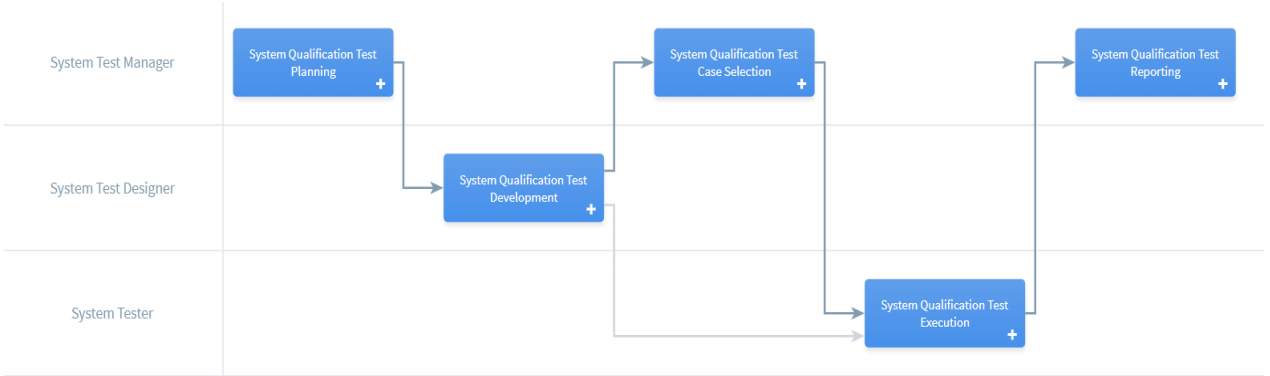
Software Product Development Workflow
Final Calibration (CAL Team) and Container generation (PEC-C)



Software Product Development Workflow
System Integration and System Qualification Tests

- System Integration

- System Qualification Tests



System Qualification Tests

- Vehicle OBD System Test
- DIAG(_S)
- OBD Scan Tool Test
- Basic Function Test System
- CHAR-Sys
- ECU Monitoring & Safety Test
- Functional System Tests
- Noise Vibration Harshness
- PA-EY Verth
- Robustness System test

Q & A

BOSCH

Thank You!

BOSCH

References

Process - Power Electronics (bosch.com)