Testing, Crafting and Developing a Brand New Powertrain Software in an Agile Environment

A joint Development around the World 🌍

Dr. Siegfried Saenger Zetina, Ralf Focken, Stefan Schlereth, Slawomir Woljnar RD Powertrain Electronics, Germany
Himank Kinkar Mercedes-Benz Research and Development, India

Mercedes-Benz

Qtronic User Conference, Dec. 2nd 2019
Who are we and what do we do?
Can the SW be ready for what comes next?
The Story

Why?
• Fit multiple powertrain variants
• Higher information quality and quantity

What?
• Create new features
• Migrate old features

How?
• Inventors, developers, testers and sponsors
• 2–3 years
SWE-Level (Software)

**MULTI-MODULE SIL**

Unit 1  
Silver Environment

Unit 2

Unit 3

Compiled Code

Simulation

**MINI COMPONENT SIL**

Engine CM

CAN-IO

Silver Environment

Engine

Compiled Code

Simulation
SYS-Level (System)

Powertrain SIL in the physical Domain for

- Torque & speed
- Current & voltage
Agile Software Development through Requirements Engineering Focus Today: Testing
How to build up awareness for testing?

1. Find the team flow
2. Start with small steps
3. Designate test-managers
4. Organize and iterate
5. Build the test concept
6. Automate everything
7. Go full variants
1. Find the Flow

- **KANBAN**
  - SYS Analysis and Synthesis
  - SWE Analysis and Synthesis

- **SCRUM 2 WEEKS**
  - Why & What
  - Architecture
  - Deployment

- **KANBAN**
  - Test & Review

- **~2 WEEKS**
  - Let’s check it
  - Be proud of it

---

Testing and Crafting Brand New Powertrain Software | Dr. Saenger Zetina | 02.12.2019
Tune the Flow

First Vehicle Integration

Summer Test-Trip
Many variants and users

2 WEEKS

NUMBER OF ISSUES

Doing Fct
Done Fct / To Do SW
Doing SW
Testing

Mercedes-Benz
Statistics

Days

- To Do: 20.83 days
- Req Writing: 10.95 days
- Architecture: 4.57 days
- Crafting: 5.39 days
- Testing: 13.79 days

Average: 4.87
Median: 2

Testing: 24%
- To-Do: 14%
- Req Writing: 35%
- Architecture: 14%
- Crafting: 13%
- Testing: 24%

To Do
14%

Crafting
13%

Architecture
14%

Req Writing
35%
2. Start with Small Steps

- Unit Construction
- Troubleshooting a Problem
- Build Testframe in Silver
- Manual Testing

**Multi-Module SIL**

<table>
<thead>
<tr>
<th>Unit 1</th>
<th>Unit 2</th>
<th>Unit 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Silver Environment</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Compiled Code
- Simulation

Stimulus
3. Designate Test-Managers

The Test-Manager works in the Agile – Team as a:

- Developer
- Product Owner
- Scrum-Master

Tasks

- Organizes Silver Hackathons
- Keeps Track of the Test-Status
- Owns the Test-Strategy
- Gets Help
4 Organize and Iterate

Test Strategy

• As efficient and as digital as possible
• Stay in one Eco-System
• Automate as much as possible

Test-Concept

• Non-Functional in Unit-Testing
• Functional in SYS-Area
• Reviews where it makes sense

Test-Tools

• SILver, Testweaver....

Decide and start testing!
5. Build the Test Concept

SYSTEM AREA

- Functional Reviews
- Acceptance Tests
- Vehicle Tests
- HIL Regression tests
- SIL Sanity Tests
- SIL Smoke Tests
- SIL-Testweaver Regression Tests

SOFTWARE AREA

- Functional Architecture Reviews
- UML-Review
- Acceptance Criteria
- Integration Tests
- Functional SIL Multi Module
- Functional SIL Unit Tests
- Non Functional Tests SIL Code Coverage
- SiL Sanity Tests
- SiL Smoke Tests
- SiL Smoke Tests
Coding test scripts from test specification is still a Manual Process
Example: Writing one Test and Matching to Requirement

# Request: a coordinated ESP request shall acknowledged to the ESP
# Test Case: if ESP request on WP#10 is confirmed, the Ack-Signal is set to EXECUTED or SUPPRESSED

# Defect:

requirement_watcher("W_CPC3_RG_TqCoor_EspReqAck_01"):
  during(Ign_Mode == 1 && Whl_StLimTq==16 && Esp_StTyp > 0):
    expect_throughout(Ptcoor_TqReqEspAck > 0 && Ptcoor_TqReqEspAck <3)

# Script to get a car started, to max speed and stop
# Script to change the powertrain configuration
6. Automate everything
Automatize the Test-Scripts

Team Mercedes-Benz RD India developed the idea💡

Scrip-Tick
Automatically generates the test scripts from test specification for various testing tools such as Testweaver, Provetech, Vector CAPL.

Advantages
• Quick Implementation
• Automated Checks on Test specifications
• Coding Tester Independent
• Standardized test specification
• Seamless integration between Test specification and test scripts
Now, how does it look?

Testconcept 2.0

Update 1: ONE Language for Script-ing

Update 2: Jenkins Pipeline

Mercedes-Benz
Sit-back and see the computer run tests

for each

SW-Release-Train the user gets a fully

• SIL-Build
• Test-Weaver Tested
• HEX-Build
7. Go full variants

Focus on 6 Main Powertrain-Release Trains (Lead Plattforms)

• Gasoline, Diesel, Mild-Hybrids, Performance Hybrids, Plug-In Hybrids, and EVs
• 4x4 and 4x2 Variants in Addition
Test-Management Today

1. Well balanced diet between SIL- and HIL-Testing
3. Next-Step: Migration of Regression Unit-Testing in SIL

- Smoke-Testing in Silver per Build
- Sanity-Testing in Silver
- Regression Testing in Testweaver
- HIL Testing
SW-Builds released and tested to users 602
Thank you very much for your attention