System Validation Process for Instrument Clusters

Thursday 24th October

Olivier BODEREAU
R&D – Instrument Clusters & Displays
System Validation Referent
Agenda

- Introduction
- System Validation Process - Test Definition
- System Validation Process - Test Development
- System Validation Process - Test Campaign
- System Validation Process - Test Follow-up & Coverage
- Conclusion
**Introduction - Magneti Marelli**

**Magneti Marelli** is an international Group (36,000 employees in 19 countries) committed to the design and production of hi-tech systems and components for the automotive sector which supplies all the most important car makers in Europe, North and South America and Asia.

**Magneti Marelli Electronic Systems** is dedicated to Instrument Clusters & Displays and Infotainment & Telematic Systems business areas.

---

**Porsche 911**

**Fiat 500**

**Citroën C4 Picasso**
Instrument Clusters & Displays development process is based on V-model in which is included System Validation.
In the early 2000s, we have defined a System Validation Process using Model Based Testing approach, with the objective of building a complete tool chain.
Test cases are written in **Doors** while **Reqtify** ensure requirements traceability & changes. **Requirements library** is an xml export from Doors that will imported in **Matelo**.
System Validation library is composed of basic modules that manage Setting & Measurement test steps.

**Test Bench library** is an xml file managed by **Matelo** that contain step descriptions & parameters.
Model design is based on test cases tagged with requirements. Transitions between 2 states contain input parameters & expected results used by associated test steps.
All projects are managed in a **Synergy** configuration tool. Basic modules are derivate from templates.
Test bench setup

Test equipment setup consists in:

- I/O setup from customer connector description
- Vision training for pattern matching & OCR
- Sound training
A test campaign is performed at each delivery to the customer. Test sequences are generated from the statistic model with several profiles.
Test campaign will be executed and test reports analyzed to check issues.

Test campaign -> Execution & Analysis
Issues are added in a **Synergy Change** database to ensure follow-up, and results are returned into **Doors** via a dxl script.

Test coverage can be evaluated by Matelo and Reqtify:

- Static coverage based on requirements present in statistic model
- Dynamic coverage based on requirements present in test sequences
Conclusion

Instrument clusters are more and more complex and evolving, and number of validations is very consistent.

This process has enabled us to reduce the duration of test campaigns thanks to:

- the generation and execution of automatic tests
- the aid in the analysis and bug tracking

Then it was possible to spend more time on model design, which gave us:

- an increasing of test coverage
- a better traceability of requirements