MANAGING QUALITY IN A VIRTUAL WORLD

Application of Quality Assurance for Virtual Vehicle Development

Dirk Frerichs
Opel Automobile GmbH
Vehicle CAE
Karlsruhe, Apply & Innovate, 12 September 2018
MOTIVATION

- System Simulation as reliable partner has to ensure **validity**
  - Requirements based validation describes the last step of a process to ensure reliance
- Models shall be used in a requirement based way
  - ‘I want to simulate…’ is not an adequate answer anymore

Goal is to establish a requirement based quality assurance for System Simulation
AGENDA

1. Term of Quality
2. Quality Activities in a Virtual World
3. Summary and Conclusion
4. Future Work
General Definition

- Quality describes the „degree to which a set of inherent characteristics of an object fulfills requirements“ (ISO 9000)

Quality in the System and Software Domain

- Defined by ISO 25000 (former ISO 9126)
- Quality of a system is the degree to which the system satisfies the stated and implied needs of its various stakeholders
QUALITY ACTIVITIES IN A VIRTUAL WORLD

Constructive Quality Assurance:
What can be done while creating software to address quality characteristics
- well-proven procedures
- well-proven components
- well-proven structures

Analytical Quality Assurance:
Analyze the software after it has been created through different testing methods
- Can be done by developers and non-developers
CONSTRUCTIVE QUALITY ASSURANCE

Base Layer

Framework Layer

Application Layer

MODEL

Model Templates
Model Config. & Param.
Base Models & Libraries
Model Conventions

System Requirements
System Architecture
Component Architecture
MIL
HIL
SIL
Acceptance Test
System Test
Integrations Test
CONSTRUCTIVE QUALITY ASSURANCE

Version control

- Git for version control for model structures, parameters and architectures
  - All versions that ever existed are accessible
  - Changes have to be explained in a commit message
  - Files are on server and on User-computer
  - Work-in-progress changes are on separate branch (Feature Branches)
  - Updates are accessible
CONSTRUCTIVE QUALITY ASSURANCE

- Version Control
- Naming Convention
- Separation of Model Structure and Parameters
- Parameter declaration file
- Parameter Process
- Model Configurator
- Standardized Interfaces
- Component Model Documentation
- Guidelines and Conventions
- Refactoring
- Model reduction
- Component Model Review
- Validation Process

Necessary fields for maintaining quality:

- Maintainability
- Usability
- Efficiency
- Security
- Portability
- Compatibility
- Functionality
- Reliability

Version Control
- Naming Convention
- Separation of Model Structure and Parameters
- Parameter declaration file
- Parameter Process
- Model Configurator
- Standardized Interfaces
- Component Model Documentation
- Guidelines and Conventions
- Refactoring
- Model reduction
- Component Model Review
- Validation Process
ANALYTICAL QUALITY ASSURANCE

Production Process → Product → Measurement → Check against Requirements

Real World

Virtual World

Adjust the process to influence quality

Characteristic A vs. Characteristic B

Min
Max
ANALYTICAL QUALITY ASSURANCE

Server for Git software projects

Script collection for automation of jobs

Tool for visualization of big data

Tool for planning and reporting
ANALYTICAL QUALITY ASSURANCE

Component Testing

Define Model and Level

Define Objective Characteristic Value

Calculate actual Value

Visualization

Level

Working Directory

GitHub

Jenkins

Jenkins Workspace

Kibana

Working Directory

Component Testing

git push Feature branch

git push Feature branch

Jenkins Job Comp.Test

Read Comp. Test Results

Visualize

Define Model and Level

Define Objective Characteristic Value

Calculate actual Value

Visualization

ANALYTICAL QUALITY ASSURANCE

Define Objective Characteristic Value

Calculate actual Objective Characteristic Value

Define limits and simulate

Steering system with Sinus Steering

Steering Sensitivity $\Lambda$

$$\Lambda = \frac{\partial a_y}{\partial \delta_H} \text{ with } \Lambda \epsilon \left[\Lambda_{\text{Min}}, \Lambda_{\text{Max}}\right]$$

Define Model and Level

System under test

- Steering system with Sinus Steering

Characteristic Value

- Steering Sensitivity $\Lambda$
ANALYTICAL QUALITY ASSURANCE
SUMMARY AND CONCLUSION

Summary

- Introduction of quality characteristics for Software Quality
  - Model for System Simulation is a software product
- Efforts to ensure quality with constructive quality assurance with a generic framework with IPG CarMaker
- Workflow for analytical quality assurance with the principle of deployment pipeline

Conclusion

- Software quality characteristics can be applied for System Simulation
  - Quality Characteristics are not decoupled and can influence each other
  - Monitoring of objective characteristics are supporting developer and user in every phase of the project
FUTURE WORK

Integrate more models...

- ... from different departments into Axiom framework
- ... into the deployment pipeline

And create a Model Factory

- All states and data of the model are traceable
THANK YOU
REFERENCES


