A layered approach for test case description in scenario-based testing and validation.

Dr.-Ing. Andreas Rauschert Dr.-Ing. Ludwig Friedmann









INTERACTION OF SOFTWARE COMPONENTS AND FILE FORMATS.



SCOPE OF THE FILE FORMATS.

Scenario

- Script of the virtual world
- "Bird view" (no vehicle internal models/states)
- Reference to map and environment model
- No evaluators

* ViL: Driving instructions and physical tests replace scenario and simulation

Test Case

- Establishment of operational readiness
- Establishment of traceability between requirement and test case
- Run scenario variation (external, optional)
- Initialization of environment simulation with concrete scenarios (external) *
- Start of (environment) simulation (external) *
- Runtime interaction (error injection, triggers) *
- Run time test evaluation *
- Post-run test evaluation *
- Storage of results
- Clean up, preparation for next run

Initialization

Execution

Tear down

Test Campaign

- List of test cases
- Distribution of test cases to test instances (SiL/HiL/ViL)
- Scheduling
- Consolidation and storage of test results

ADVANTAGES OF THE LAYERED APPROACH.

- Scenarios are exchangeable and re-usable throughout test instances and test cases.
- Test cases abstract requirement-based tests from test instance specific architecture, enabling consistency throughout SiL/HiL/ViL.
- Test campaign defines distribution and evaluation of large numbers of test cases on test instances.
- Modular software architecture enables testing of requirements that do not impose scenario-based testing or scenario-variation.

Questions & Answers



