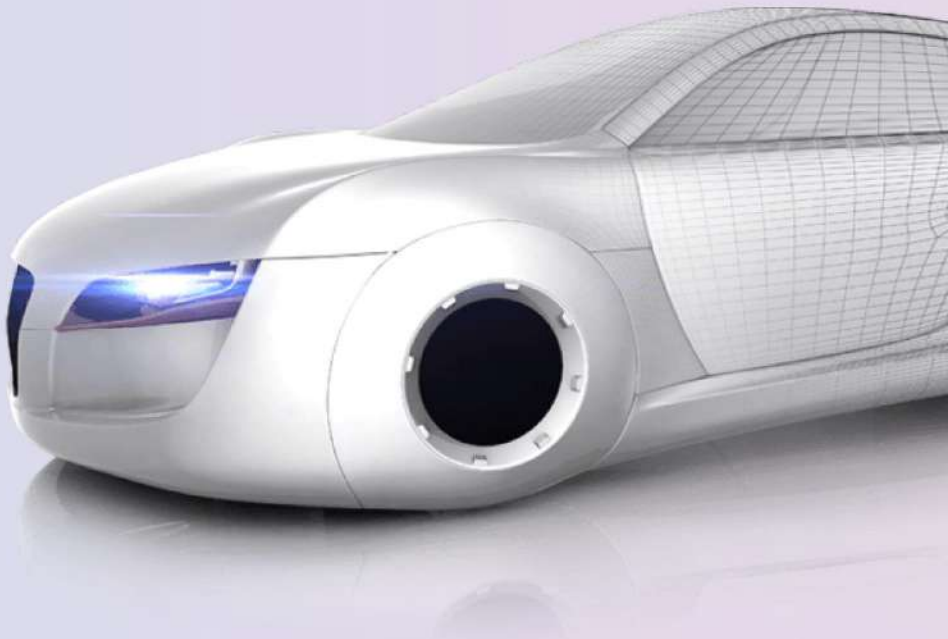
DUBAI WORLD CONGRESS
FOR SELF-DRIVING TRANSPORT
OCT | 2019

Verification & Validation approaches for Highly Automated Driving (HAD)



Plato Pathrose

Head of Validation ADAS & Autonomous Driving

Visteon Electronics Germany GmbH

Agenda

- Introduction & Overview – Visteon as Tier 1
- Agile & Scenario based Testing
- Driver in Loop Testing
- Vehicle Testing
- Test Classification & Acceptance Criteria

Introduction & Overview
Visteon as Tier 1 in ADAS & Autonomous Driving

Visteon in ADAS & Autonomous Driving

Failsafe Secure Centralized Computation

OPEN for algorithm developers

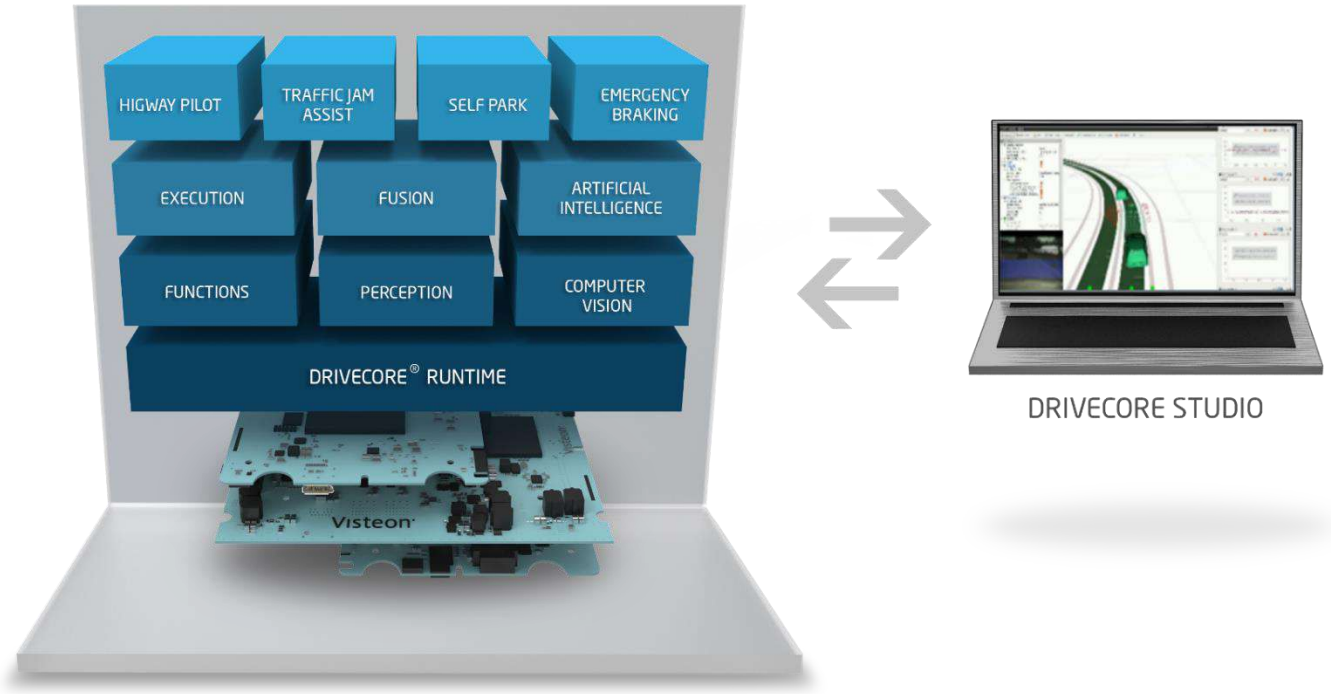
Artificial Intelligence in Autonomous Driving

DriveCore™ enables fast, controlled and secured algorithm development for autonomous driving

DriveCore™

Visteon's Autonomous Driving Platform

- DriveCore™ Studio: OPEN for Common Development
- DriveCore™ Compute: Modular Design with SoC Flexibility
- DriveCore™ Runtime: Safe Communication and Easy Integration in Vehicle



drivecore™

Compute | Runtime | Studio

Visteon's DriveCore™ Autonomous Driving Product

Visteon Components

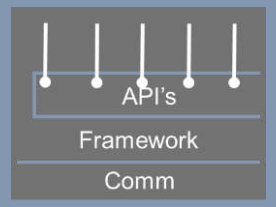
Visteon Offerings

Development Environment

DriveCore™ Studio

Cloud and PC-based tool for algorithm development:

- Sensor data record/playback
- Visualization, configuration, testing, comparison, validation and versioning
- Simulated Environments; Eclipse IDE integration

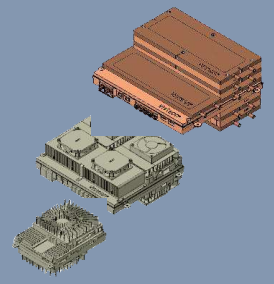


Middleware

DriveCore™ Runtime

Middleware for high-frequency/low-latency communications:

- Including APIs
- Layer for sensor data and integration of algorithms
- Framework for algorithms
- Easy integration of 3rd party software and algorithms



Hardware Platform

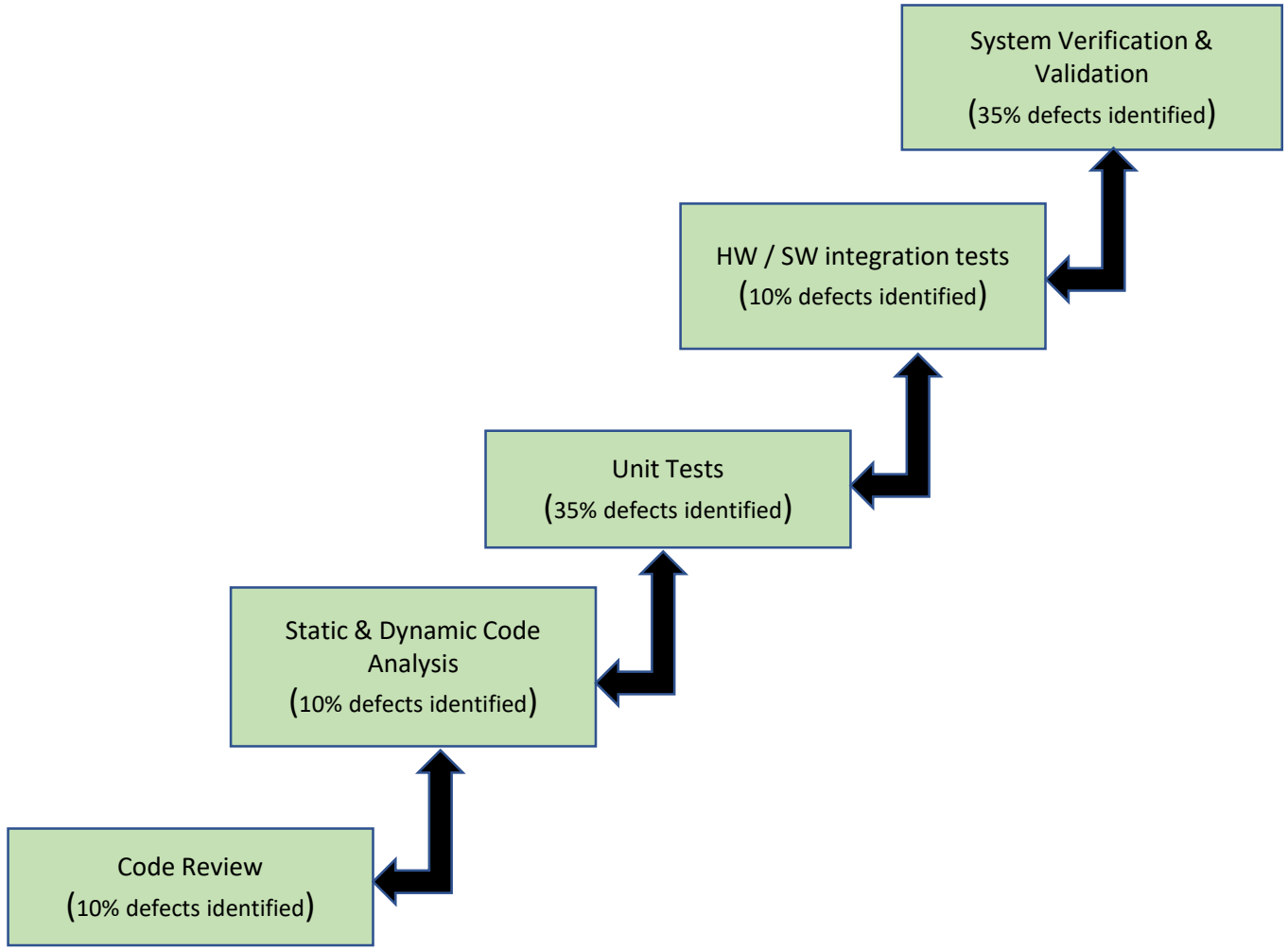
DriveCore™ Compute

Scalable computing hardware platform with ASIL compliance:

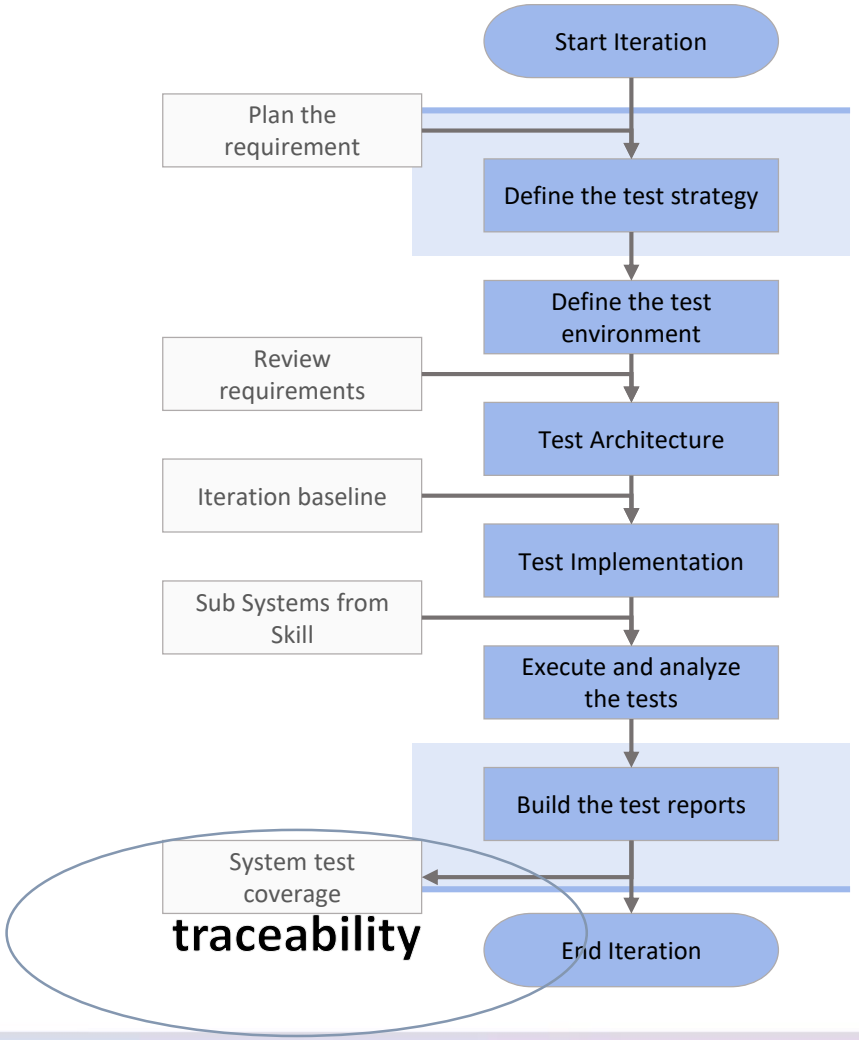
- Multi SoC design & flexibility
- From Level 2 to Level 4
- Failsafe, fault tolerant scalable & modular computing
- ASIL compliance

Agile & Scenario based Testing

Test Strategy & Process to the Lowest Level – V Model Approach



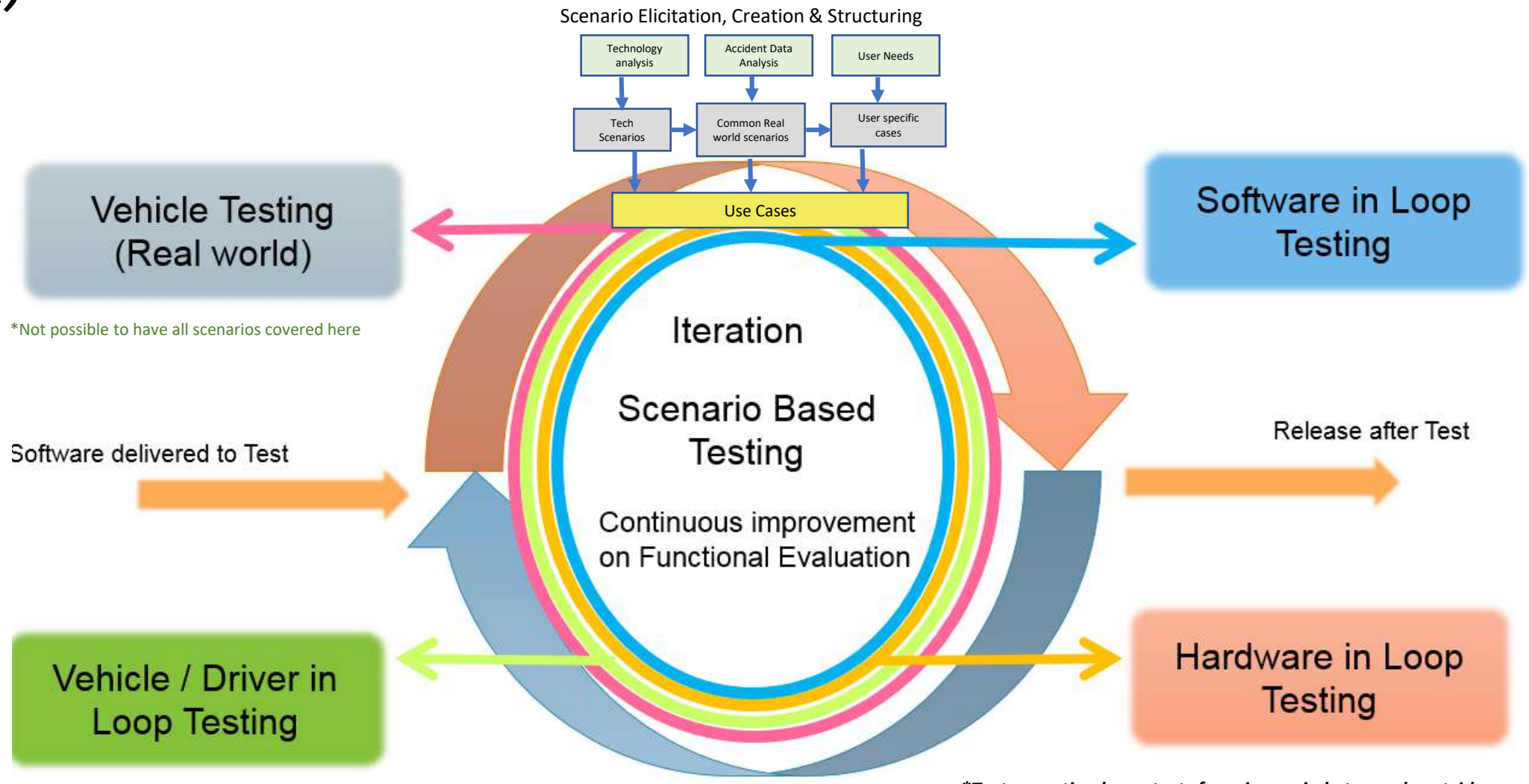
Test Process Flow



Why Agile Verification over Classical V-model Process ?

- V-Model requires all requirements in early phase of the development.
- ADAS / Autonomous Systems being complex, cannot cover all requirements at the early phase of development.
- Normal development time period of the product is approx. 3 yrs or more (End product deployment)
- Since having incremental requirements, it is not possible to execute verification at the end of the Product development.
- Tests to be planned based on Scenarios rather than focusing on millions of kms drive alone.
- Scenarios keeps on incrementing for testing hence continous improvement of test suite and scenario based risk assesment (Monte-Carlo Analysis) to be performed.

Scenario Based Testing for Level 3 features (Case study: Highway Pilot)



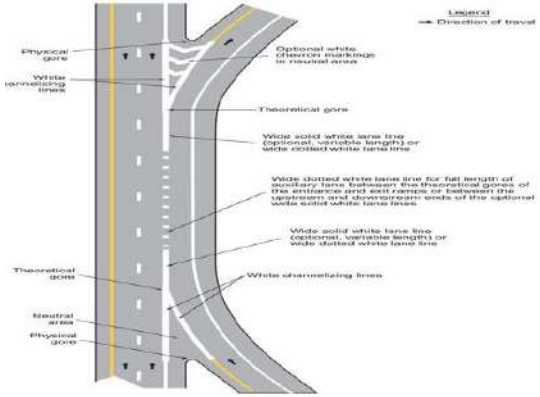
Test execution loop starts from inner circle towards outside

Scenarios were generated based on following scenes

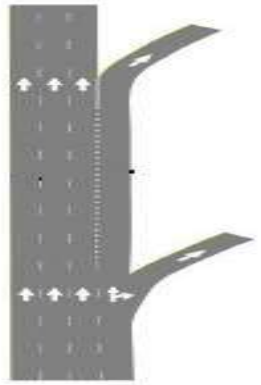
1. Ramp Assistance
2. System Start Up (Autonomous driving start up)
3. Lane Keeping
4. Driving to main lane and out (joining and exiting highway)
5. Lane change
6. Road Maintenance scenarios
7. Congestion (Traffic situation in Highway)
8. Tunnel Driving
9. Exist & Entry to Highways

Different scenarios for operating conditions Driving on the Exit and Entry Ramp for Joining and exiting the Highways

1.a Merging of two entry Ramps on A highway



2.a Entry Ramp followed by Exit Ramp



3.a Parallel Exit Ramps

1.b Merging of two entry Ramps on A highway

2.b Entry Ramp followed by Exit Ramp

3.b Parallel Exit Ramps

1.a, 2.a, 3.a are a few of the real world scenarios for Ramp Driving which are implemented in the virtual environment in 1.b, 2.b and 3.b respectively

Parameterization

Requirements – Feature under test

Scenario Strategy

Create Scenarios

Single Scenario

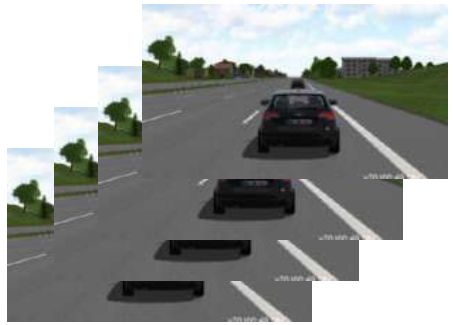
Identify Variable Parameters in Scenario

Controllable Parameters

All combinations identified and cross combinations applied

Prioritize, Vary and Generate Combination of Parameters Range Conditions in a Scenario

Scenario based Test Execution for Every Single Combination



Generated Multiple Situations

Multiple Situations from same scenario for Test Runs

Test Execution (Full / Regression)

Use Case

The automatic generation of the variations for the Test scenario Entry Ramp followed by an exit ramp can be explained using the images below :

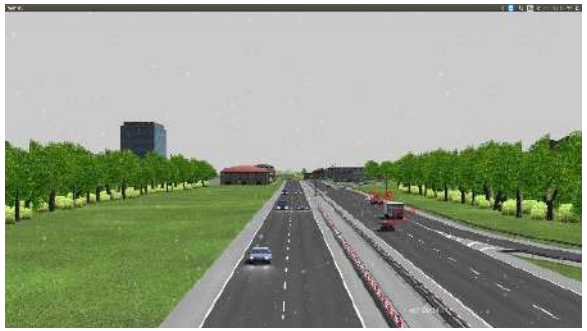
Day Time

Dawn

Dusk

Night Time

With Clouds



With Snow

With Rain Overcast

With Rain and Fog

Above are a few variations generated automatically using the ASG concept wherein for the test scenario “Entry ramp followed by an exit ramp” the following parameters were varied for this example:

- Time of the day within a specified range
- Weather conditions like snow , rain with different intensities and combination of both
- Friction of the road within a specific range

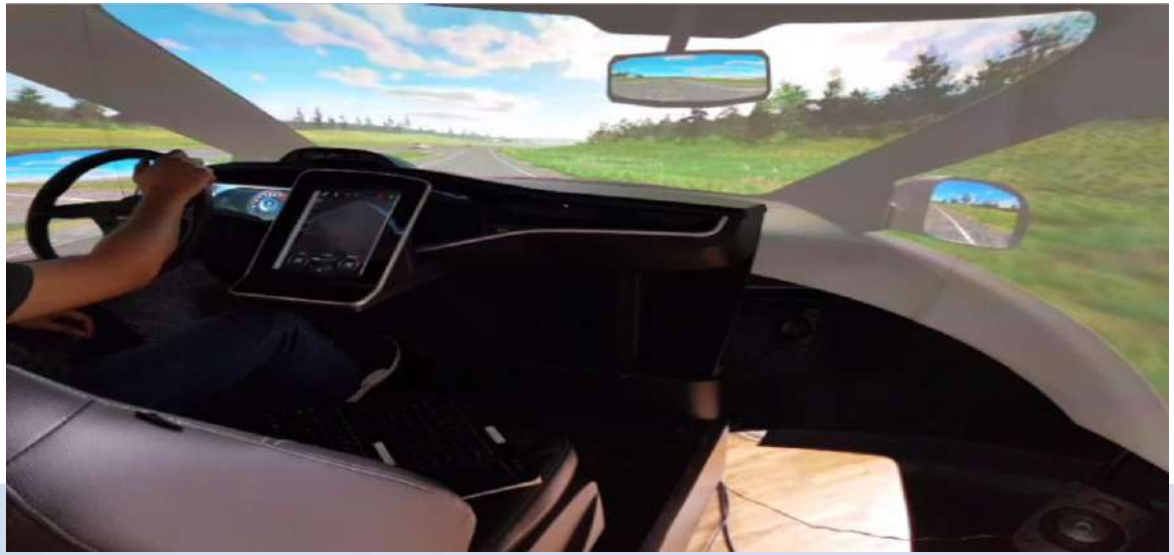
etc..

There are many other such parameters which can be automatically varied to cover the multidimensional space of the real scenarios .

Driver-in-Loop Testing

- Mimic real vehicle functionality and associated human factors
- High degree of realism of driving environment
- Immersive and realistic Driver-in-Loop (DIL) simulation experience
- Control the Car with steering wheels and pedals
- Provides better understanding of vehicle dynamics
- Vehicle control systems similar to ones on the real roads

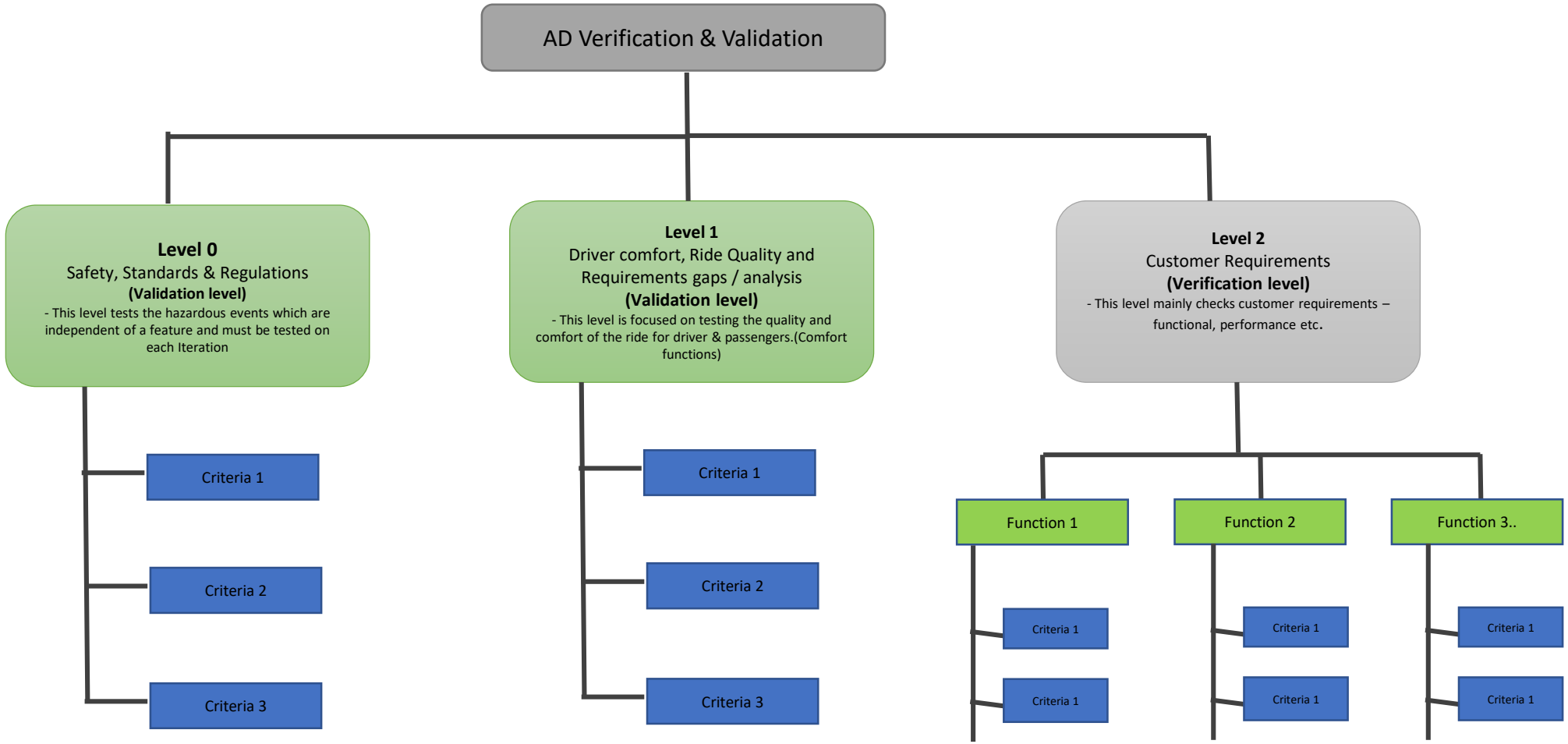
Visteon Vehicle Simulator



Vehicle Testing Process (each iteration)

SiL Testing is finished	•System Integration is finished •No critical defects •Software is stable and backward compatible
Open Loop	•Open loop tests are done •No safety related defects present
Test Track/Proving Ground-Closed loop Testing	•Functional+ System testing is done •Less then 20% defects present (than lab environment) •No safety related defects + required closed loop mileage is achieved e.g. 1000kms •Defect Criticality Analysis is done for public road testing Go/NoGo decision
Public Road Testing-Closed loop	•Low/medium/High Speed test routes for each feature(low, moderate and high traffic situations) •Functional and System verification is done •Defects reported and tracked •No safety related defects + Required closed loop mileage is achieved e.g. 1000kms
Validation Release	• Test reports + Test results published • Defects are reported and tracked • Less then 2% non critical defects present(comparison with lab based test results)
SW Release to Customer	• Software is released • Release notes, results reports and defect lists Provided

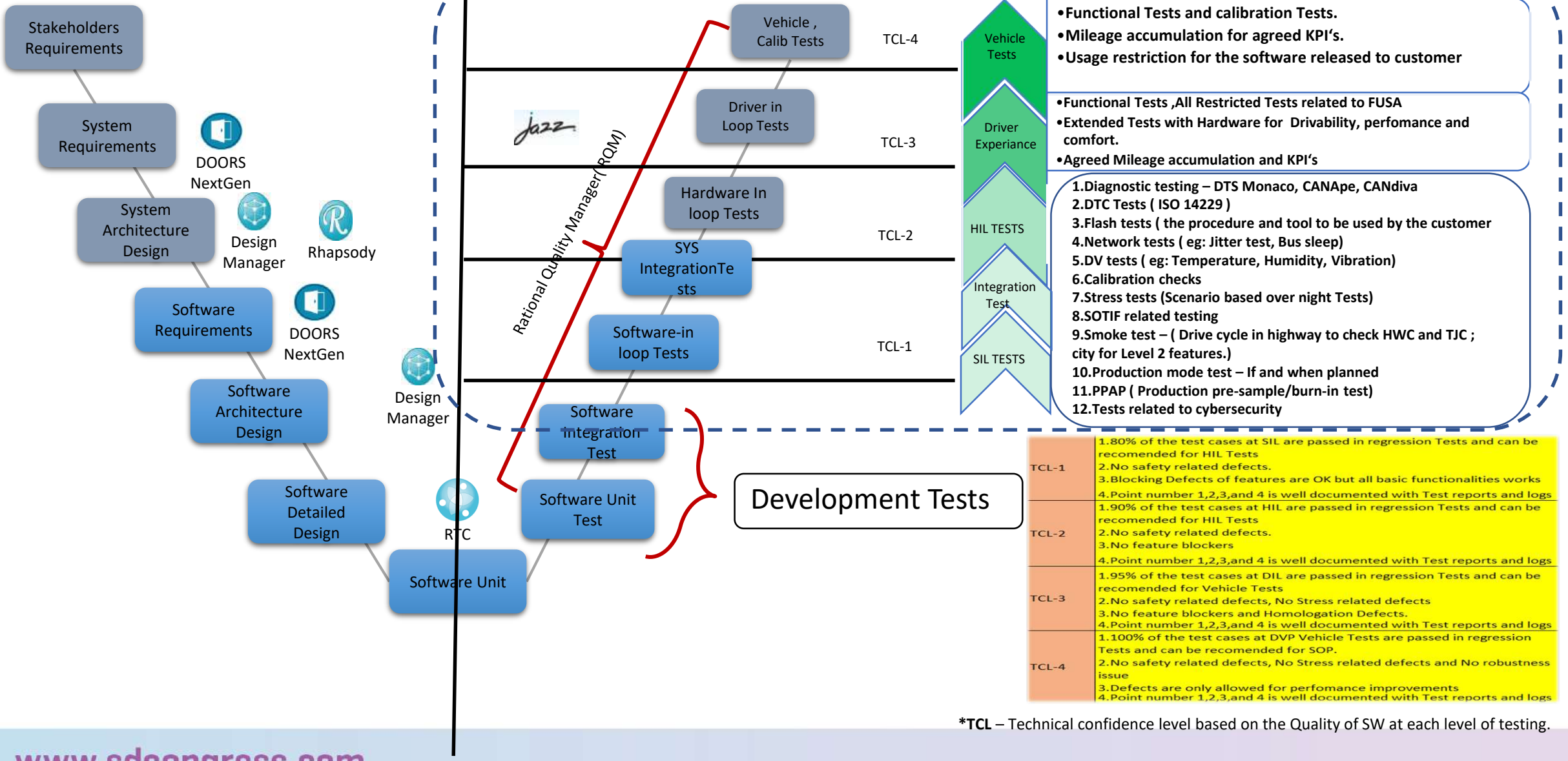
Verification & Validation Classification



Validation approach involves 3 levels starting from Level 0 and Level 1 which checks in the perspective of validation and executed based on standards and regulations and also internal expertise in Visteon. Level 2 is verification which mainly focus on the customer requirements.

Level 0 is also considered as the high priority tests which decides the system / software is good to be taken forward for further elaborated testing.

Verification & Validation



Development Tests

TCL-1	1.80% of the test cases at SIL are passed in regression Tests and can be recomended for HIL Tests 2.No safety related defects. 3.Blocking Defects of features are OK but all basic functionalities works 4.Point number 1,2,3,and 4 is well documented with Test reports and logs
TCL-2	1.90% of the test cases at HIL are passed in regression Tests and can be recomended for HIL Tests 2.No safety related defects. 3.No feature blockers 4.Point number 1,2,3,and 4 is well documented with Test reports and logs
TCL-3	1.95% of the test cases at DIL are passed in regression Tests and can be recomended for Vehicle Tests 2.No safety related defects, No Stress related defects 3.No feature blockers and Homologation Defects. 4.Point number 1,2,3,and 4 is well documented with Test reports and logs
TCL-4	1.100% of the test cases at DVP Vehicle Tests are passed in regression Tests and can be recomended for SOP. 2.No safety related defects, No Stress related defects and No robustness issue 3.Defects are only allowed for perfomance improvements 4.Point number 1,2,3,and 4 is well documented with Test reports and logs

***TCL** – Technical confidence level based on the Quality of SW at each level of testing.

Thank You !

&

Questions ?