COMPOSE VIRTUAL CARS

A Semi-Automated Approach of Composing Virtual Vehicles in Projects with Large Numbers of Vehicle Variants

Dr.-Ing. Matthias Borsdorf
Adam Opel AG, Control Systems & Brake CAE

Karlsruhe, Apply & Innovate, 20 September 2016
AGENDA

1. Tool Chain Overview
2. Motivation
3. Handling of CarMaker Vehicles
4. Composer Concept & Process
5. Summary and Conclusion
MOTIVATION

Variants with Controls Relevance in a Vehicle Program

- Engines
- Transmissions
- Suspensions
- Chassis setups
- ACC types
- ESC hardwares
- Park brakes
- Start/Stop options
- Camera options

~ 100 variants

Update of one parameter

Several vehicles to be updated
HANDLING OF CARMAKER-VEHICLES

- Update of parameters hard to handle with GUI
- ASCII file usage enables Copy & paste among vehicle variants

Need for a tool to set up all vehicle variants

- Efficient
- Reliable
- Repeatable

„Composer“
COMPOSER CONCEPT

- Split ASCII-infofile into several component ASCII-files
- Include parameters for Simulink-model in component ASCII-files
- Combine component ASCII-files together to whole vehicle infofiles
VEHICLE DESCRIPTION BY RPO CODES

- Coding system from production (Regular Production Option)
- One RPO has 3 alphanumeric characters
- A vehicle variant → completely described by a list of several RPOs
  
  \[\text{vehicle description}(\text{Vehicle1}) = \text{AAA} & \text{BBB} & \text{CCC} & \text{DDD} & \text{JJJ}\]

- Each part has a list of RPO required in a vehicle to be used
  
  \[\text{usage statement (Part 111111)} = \text{AAA} & \text{JJJ}\]

This vehicle description system is also used for virtual vehicles

<table>
<thead>
<tr>
<th>RPO for Brakes</th>
<th>Option Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JJJ</td>
<td>FRT DISC 14inch, RR DRUM 8inch</td>
</tr>
<tr>
<td>JJQ</td>
<td>FRT DISC 15inch, RR DISC 15inch</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Vehicle description (Vehicle1) = AAA & BBB & CCC & JJJ

- Does part 111111 or part 111112 belong to this vehicle?

  usage statement (part 111111) = AAA & JJQ
  usage statement (part 111112) = AAA & JJJ

<table>
<thead>
<tr>
<th>RPO for Brakes</th>
<th>Option Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JJJ</td>
<td>FRT DISC 14inch, RR DRUM 8inch</td>
</tr>
<tr>
<td>JJQ</td>
<td>FRT DISC 15inch, RR DISC 15inch</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
COMPOSER PROCESS

1. Create vehicle variant table
2. Create component ASCII infofile for each component
3. Execute composer: combine component-ASCII-files into a lot of vehicles
4. Verify data format: open & save in Carmaker (automated)
5. Parametercheck: validate same behavior as in Multi Body Simulation
6. Upload vehicles and distribute to users
# VARIANT TABLE

<table>
<thead>
<tr>
<th></th>
<th>Vehicle001</th>
<th>Vehicle002</th>
<th>Vehicle003</th>
<th>Vehicle004</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>AAA</td>
<td>AAA</td>
<td>AAA</td>
<td>AAA</td>
</tr>
<tr>
<td>Engine</td>
<td>BBB</td>
<td>BBB</td>
<td>BBB</td>
<td>BBB</td>
</tr>
<tr>
<td>Transmission</td>
<td>CCC</td>
<td>CCH</td>
<td>CCW</td>
<td>CCH</td>
</tr>
<tr>
<td>Brakes</td>
<td>JJJ</td>
<td>JJJ</td>
<td>JJQ</td>
<td>JJQ</td>
</tr>
<tr>
<td>Region</td>
<td>EUR</td>
<td>EUR</td>
<td>EUR</td>
<td>EUR</td>
</tr>
</tbody>
</table>

Vehicle description(Vehicle001) = AAA & BBB & CCC & JJJ & EUR
Vehicle description(Vehicle002) = AAA & BBB & **CCH** & JJJ & EUR
CREATE COMPONENT

- ASCII file for each component

Usage statement: model & brake

Description goes to

## Additional Parameters (Misc)

CM parameters

Parameters for MasterModel

Simulink Environment

```plaintext
### Brake System
Composer.RPO.1 = AAA & JJJ
RPO.Brake = JJJ
OptionDescription.Brake = FRT DISC 14inch, RR DRUM 8inch

### CarMaker Brake Parameters
Brake.Park.Torque_max = 0 0 1000 1000
Brake.Torque.Amplify = 1 1 1 1
Brake.Kind = PresDistrib 1
Brake.pMC_based_on = PedalAct
Brake.Pedal2PedalFrc = 1
Brake.PedalAct2pMC = 200
Brake.PedalForce2pMC = 1
Brake.tResp = 0.05
Brake.tBuildUp = 0.2
Brake.pWB2Trq = 25 25 12 12
UsrCfgVeh.Brk.PedalRatio = 4
UsrCfgVeh.Brk.ForceJumpIn = 400
UsrCfgVeh.Brk.BoostRatio = 8
UsrCfgVeh.Brk.e_BoosterType = 5
```
EXECUTE COMPOSER

For each vehicle:
Composer selects component files

Composer combines selected component files

CM vehicle infofile

vehicle1 description
AAA & BBB & CCC & JJJ &...

Composer Database

parameters

All components

Component – files for vehicle 1

Component – files for vehicle 2

vehicle1

vehicle2 description
AAA & BBB & CCH & JJJ &...
VERIFY DATA FORMAT

Ensure created vehicles fit the CM infofile format
→ script opens the composed infofile and saves it with CarMaker.

```
Report of differences between cm-files
Car: 
 "AAA_BBB CCC JJJ 042.car"
Modified car: 
 "AAA_BBB CCC JJJ 042_modifiedByCM.car"
Number of keys differs!
------------------------
Those keys are missing: 
"AAA_BBB CCC JJJ 042.car"
"AAA_BBB CCC JJJ 042_modifiedByCM.car":
   Aero.Comment
   Susp.F.SecSpring.Amplify
   Susp.R.SecSpring.Amplify
   PowerTrain.GearBox.Conv.I_in
   PowerTrain.GearBox.Conv.I_out
   PowerTrain.GearBox.Conv.mue
------------------------
The following keys differ: 
   FileCreator
```

- CarMaker adds default values
- CarMaker adds unnecessary values: No need of converter in manual transmission
- Change composer
- Reported to IPG / Do not care
- File creator changes during saving
- As expected
- Nothing else
- great
PARAMETERCHECK

- Validate same behavior as in Multi Body Simulation (MBS)
- Ensure quality of created vehicles

Multi Body Simulation (MBS) Master Model

CM Vehicle infofile

Control Systems Master Model

Simulation

MBS results of specific maneuvers

MBS results as comment in vehicle file

CM results of specific maneuvers

Compare

Assessment
UPLOAD VEHICLES AND DISTRIBUTE

- Communicate quality and limitations
- Users with different use cases
- Used for regular vehicle development

Use cases

- Algorithm development
- Testing
- Calibration
CONCLUSION AND OUTLOOK

• Composer → efficient, reliable & repeatable creation of vehicle variants

• Semi-automated process → Fully automated creation of ’variant list’ and ‘usage description’ from physical vehicle RPO-database - a challenge

• Growing complexity & numbers
  • Vehicle variants
  • Users
  • Maneuvers (e.g. NCAP)
  • Methods (HiL, SiL, MiL, PiL, DiL)
  • Model complexity
  • Usage (algorithm development, calibration, testing)