SIMPACK - A Tool for Off-Line and Real-Time Simulation

- Real-Time for ECU Testing: State of the Art and Open Demands
- SIMPACK - Code Export: A Newly Emerging Module for Real-Time Models
- Application Example

Coming from the Off-Line World ...

Single Source Data Management

Digital Mock-up

- System Assembly
- Fly Through
- Motion Envelops
- Collisions
- ...

Functional Virtual Prototyping (MBS)

- Kinematics
- Handling
- Stability / Derailment
- Ride Comfort
- Noise / Vibration / Harshness
- Durability / Fatigue
- Safety / Impact
- System/Track Interactions

Virtual Production

- Tolerance
- Assembly
- Sequencing
- ...

... to meet Real-Time World of ECUs

- **ABS**
- **TCS**
- **ESP**
- **VSC**
- **ACC**
- **EBA**
- **ABC**
- **VDC**

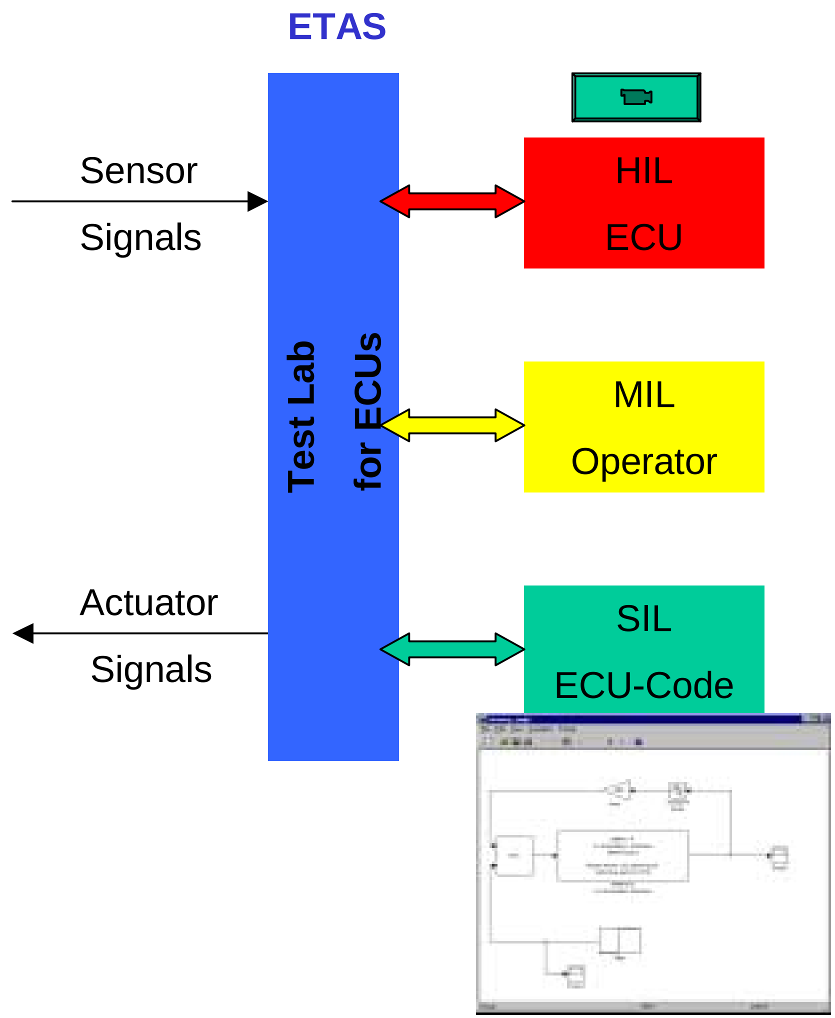
Functional Virtual Prototype

<u>MBS-Model (1-2 ms)</u>

- Wheels / Tyres
- Suspensions
- Steering System
- Chassis
- Brakes
- Driver
- Road
- ...

<u>Actuators (0.5 ms)</u>

- Hydraulics
- Electronics
- Pneumatics
- ...



What does Real-Time Simulation mean to the FVP ?

Functional Virtual Prototype

<u>MBS-Model (1-2 ms)</u>

- Wheels / Tyres
- Suspensions
- Steering System
- Chassis
- Brakes
- Driver
- Road
- ...

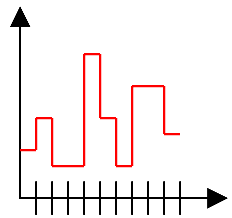
$$\dot{x} = f(x,u,t)$$

$$y = g(x,u,t)$$

<u>Actuators (0.5 ms)</u>

- Hydraulics
- Electronics
- Pneumatics
- ...

Sensor Signals y →



← Actuator Signals u

- It always means „Co-Simulation“ with a fixed sample rate
- It means no “Jacobian” of Complete System
- It means within 0.5 - 2 ms: - at least one function evaluation
  - at least one integration step
  - one sensor evaluation
- Today it mostly means (semi) explicit Euler scheme for FVP
- Today it means ordinary differential equations
- Today it means non-stiff systems only
- Today it means no kinematic closed loops (DAEs)
- Today it means simple models

Approaches Today to Meet Requirements and Limitations

Approaches Today

- Hand coded, highly tuned, simple MBS models
- Representing kinematics of suspension by (multi-) dimensional look-up tables
- Adapting solver to topology and elements of each individual MBS
- Co-Simulation of MBS and/or actuator models
- Combination of methods

Limitations

- High implementation effort whenever system changes
- Low compatibility with existing off-line models for handling, ride, durability, ...
- Difficult process save parameterisation when using look-up tables
- Break in process line from off-line world to real-time world
- Nearly no flexible body support (stabiliser, leaf-spring, twist-beam axle, sub-frame)?

Open Demands

fast at a fixed sample rate , stable, highly detailed, automatic, process save

- Automatic generation of real-time models from existing FVPs using existing parameterisation, data pool and sub-structuring
- Process save model reduction techniques avoiding (multidimensional) look-up tables
- Library with different levels of detail for sub-systems like suspensions, steering systems, drive trains,
- Identification environment for finding the parameters of a reduced real-time model
- Account for flexible body components
- Real-time solvers for numerically stiff systems and differential algebraic systems
- Partial linearisation and multi-linear systems
- Stable and ease-of-use co-simulation of MBS and/or actuator models to support a multi-processor approach

SIMPACK - A Tool for Off-Line and Real-Time Simulation

- Real-Time for ECU Testing: State of the Art and Open Demands
- SIMPACK - Code Export: A Newly Emerging Module for Real-Time Models
- Application Example

SIMPACK Symbolic Accelerator

Equations of Motion $\dot{x} = f(x,u,t)$

Sensor Equations $y = g(x,u,t)$

SIMPACK Symbolic Accelerator

Step 0 (once): Load Model

Step 1 (once):

Generate algorithm for $\boldsymbol{f}$ and $\boldsymbol{g}$ of an individual MBS

- eliminate all if, else, do, ... operations
- perform all 0, 1 and const. operations
- (perform partial linearisation)
- (perform additional model reductions)
- recursively check code for non-used statements

$\Rightarrow$ results in highly efficient code

Step 2 (once):

Compile and generate executable

Step 3 (in each integration step):

Evaluate **optimised** code for $\boldsymbol{f}$ and $\boldsymbol{g}$

Numerical Codes (SIMPACK, ADAMS, ...)

Step 0 (once): Load Model

(in each integration step):

Evaluate **generic** code for $\boldsymbol{f}$ and $\boldsymbol{g}$

SIMPACK Symbolic Accelerator - Fields of Application

✓ Reduce simulation time for

SIMPACK- Virtual Test Lab (Parameter Variation, DoE, Batch)

SIMPACK - Optimisation

How SIMPACK Code Export Works

Equations of Motion $\dot{x} = f(x,u,t)$

Sensor Equations $y = g(x,u,t)$

SIMPACK Symbolic Accelerator

Step 0 (once): Load Model

Step 1 (once):

Generate algorithm for f and g of an individual MBS

- eliminate all if, else, do, ... operations
- perform all 0, 1 and const. operations
- (perform partial linearisation)
- (perform additional model reductions)
- recursively check code for non-used statements

$\Rightarrow$ results in highly efficient code

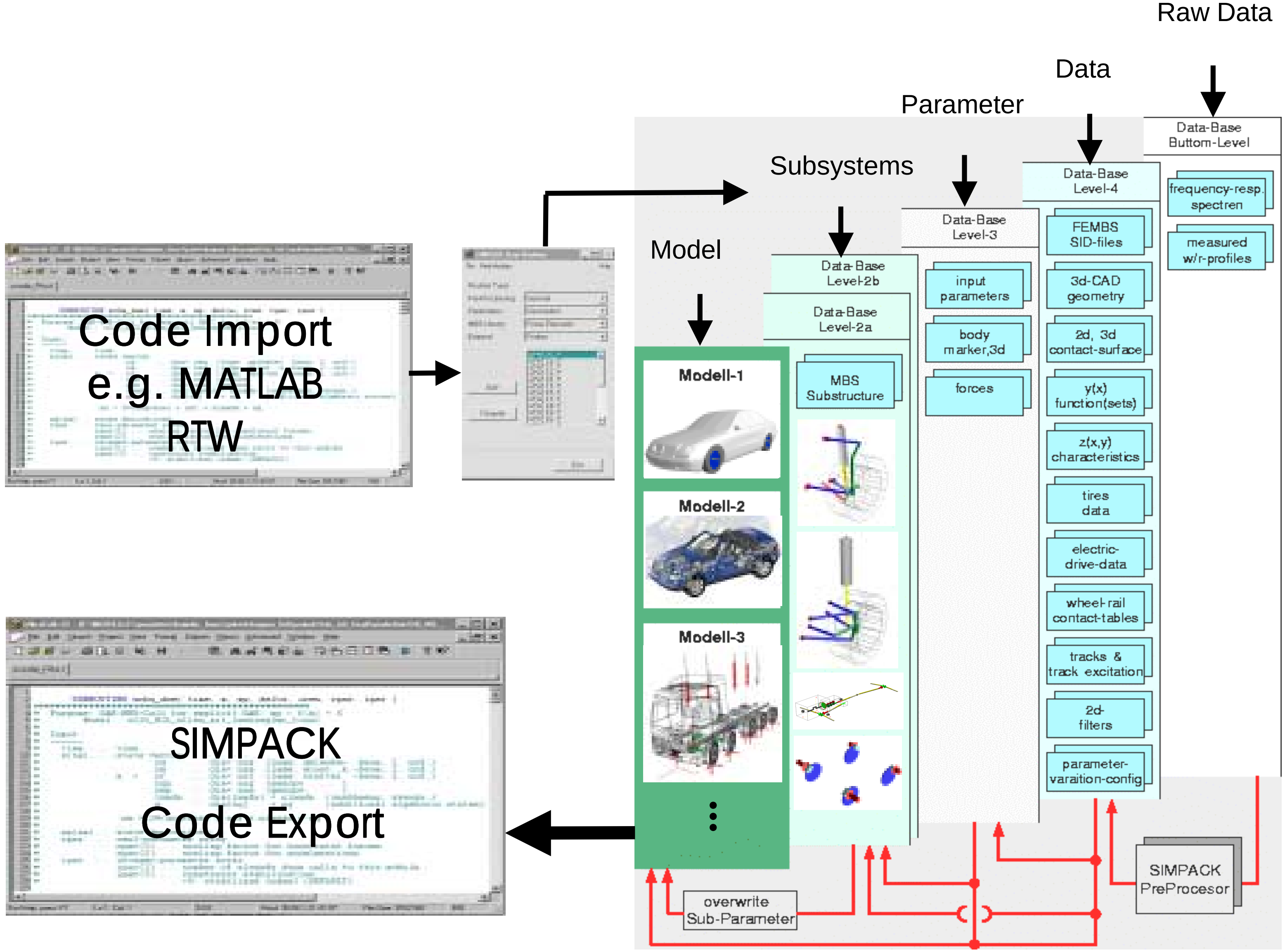
SIMPACK Code Export

Step 2 (once):

- adapt dimensions to problem size
- resolve all SIMPACK dependencies
- export code to external simulation environment
-

Ongoing Development

SIMPACK Code Export



SIMPACK Code Export - Fields of Application

- ✓ Automatic generation of real-time models for ECU development and testing
- ✓ Automatic Generation of ETAS LabCar models
- ✓ Plug-in SIMPACK-model (and -solver) for MATLAB/Simulink
- ✓ Plug-in SIMPACK-model (and -solver) for ASCET-SD
- ✓ Plug-in SIMPACK-model (and -solver) for AMESIM
- ✓ Plug-in SIMPACK-model (and -solver) for any external simulation environment
- ✓ Archive simulation scenario independent of SIMPACK

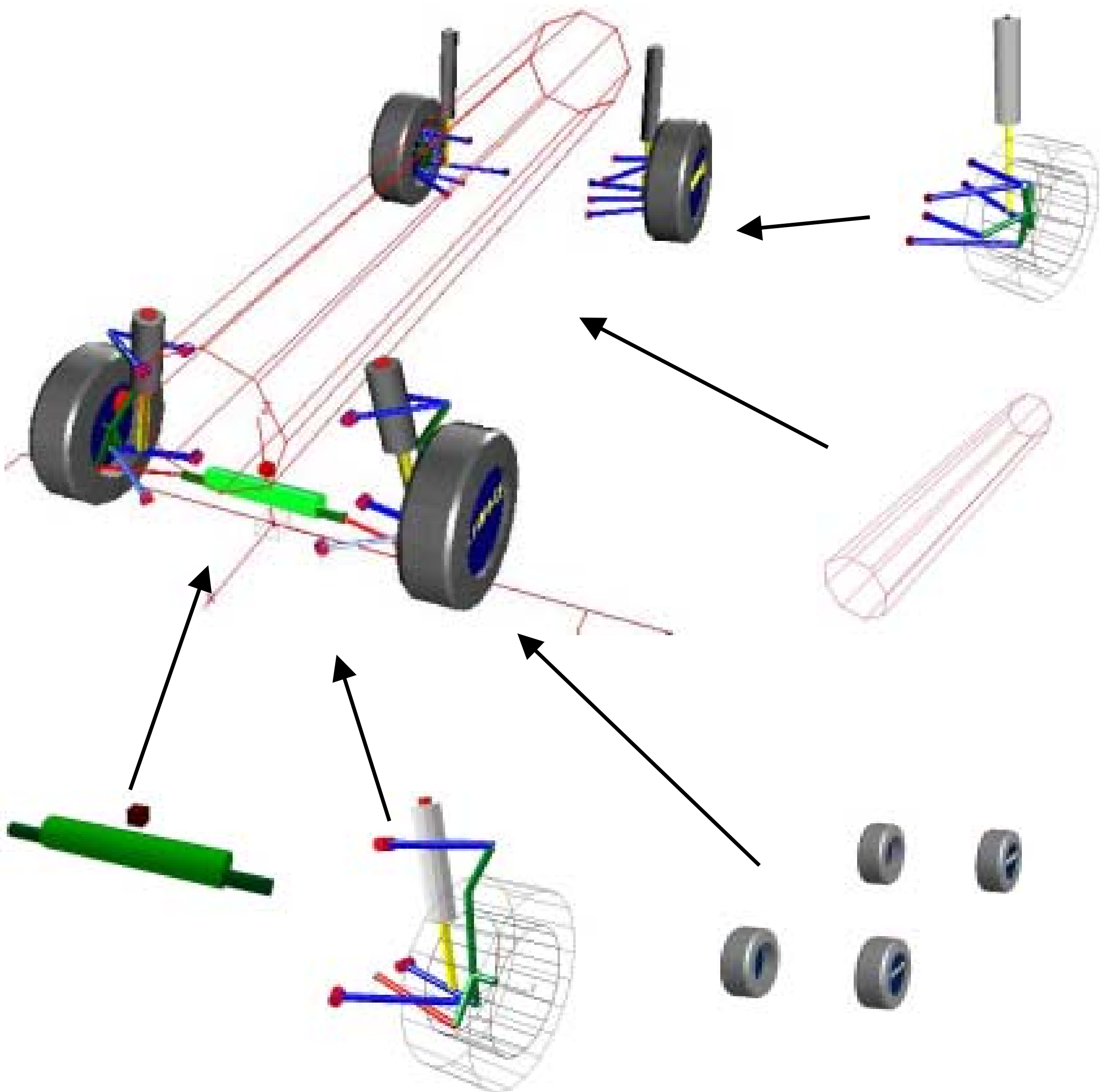
Contribution of SIMPACK Code Export to Real-Time Models

- ✓ Automatic generation of real-time models from existing FVPs using existing parameterisation, data pool and sub-structuring
- ✓ Library with different levels of detail for sub-systems like suspensions, steering systems, drive trains
- ✓ Account for flexible body components
- ✓ Co-simulation of MBS and/or actuator models
- ✓ CPU-time reduction by partial linearisation
- ✓ CPU-time reduction by SIMPACK Symbolic Accelerator
- ✓ CPU-time reduction by model reduction techniques avoiding (multidimensional) look-up tables

SIMPACK - A Tool for Off-Line and Real-Time Simulation

- Real-Time for ECU Testing: State of the Art and Open Demands
- SIMPACK - Code Export: A Newly Emerging Module for Real-Time Models
- Application Example

Application Example



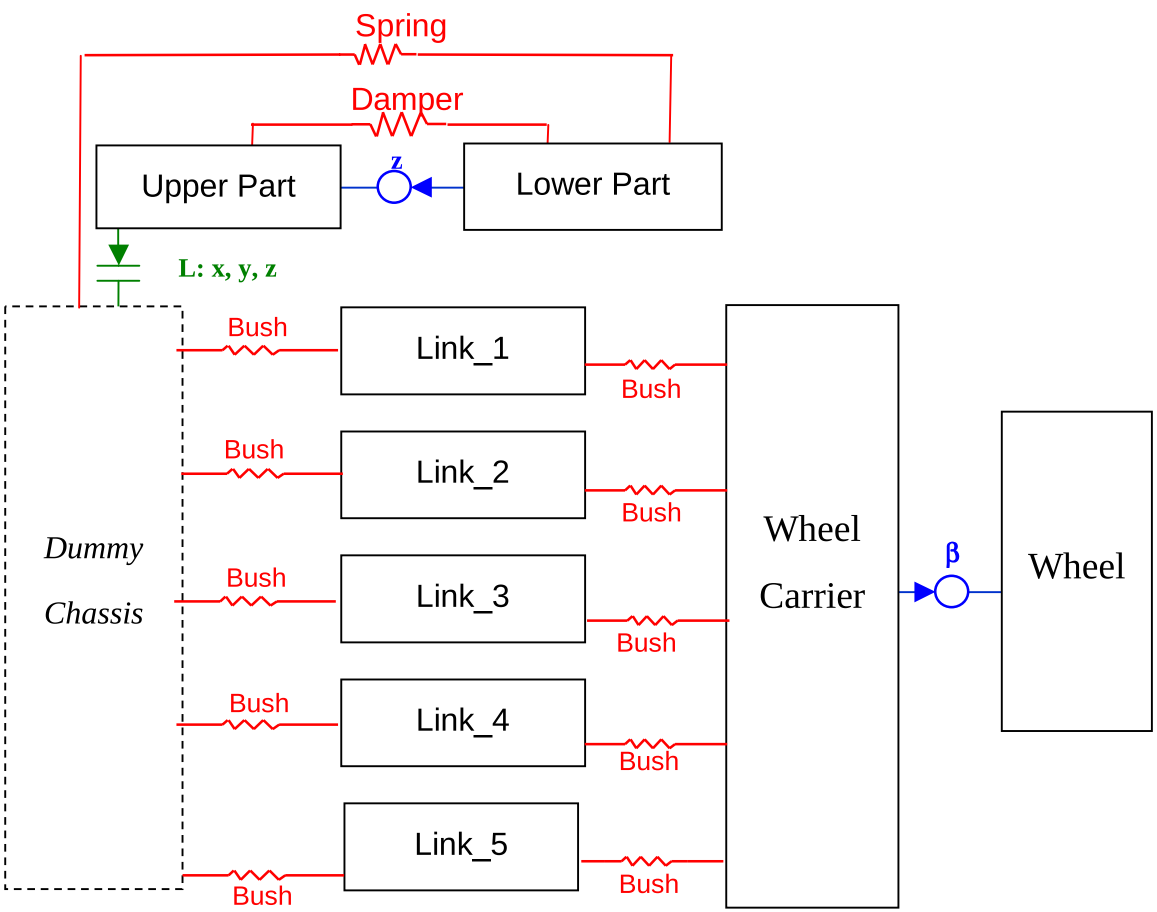
Full SIMPACK Vehicle Model:

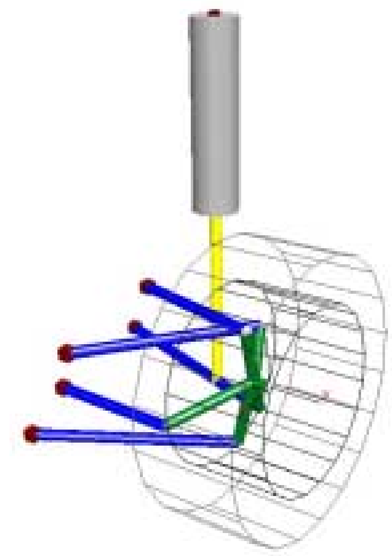
based on sub-structure technique:

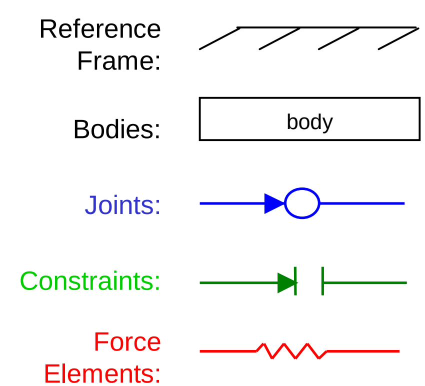
- Rear axle with full compliance
- Front axle with full compliance
- Chassis
- Steering System
- Pacejka Tyres
- Fully parameterised
- 10 sec sinusoidal steering manoeuvre
- 800 MHz Intel CPU

Application Example: Reduction of Rear Suspension

Full SIMPACK Model

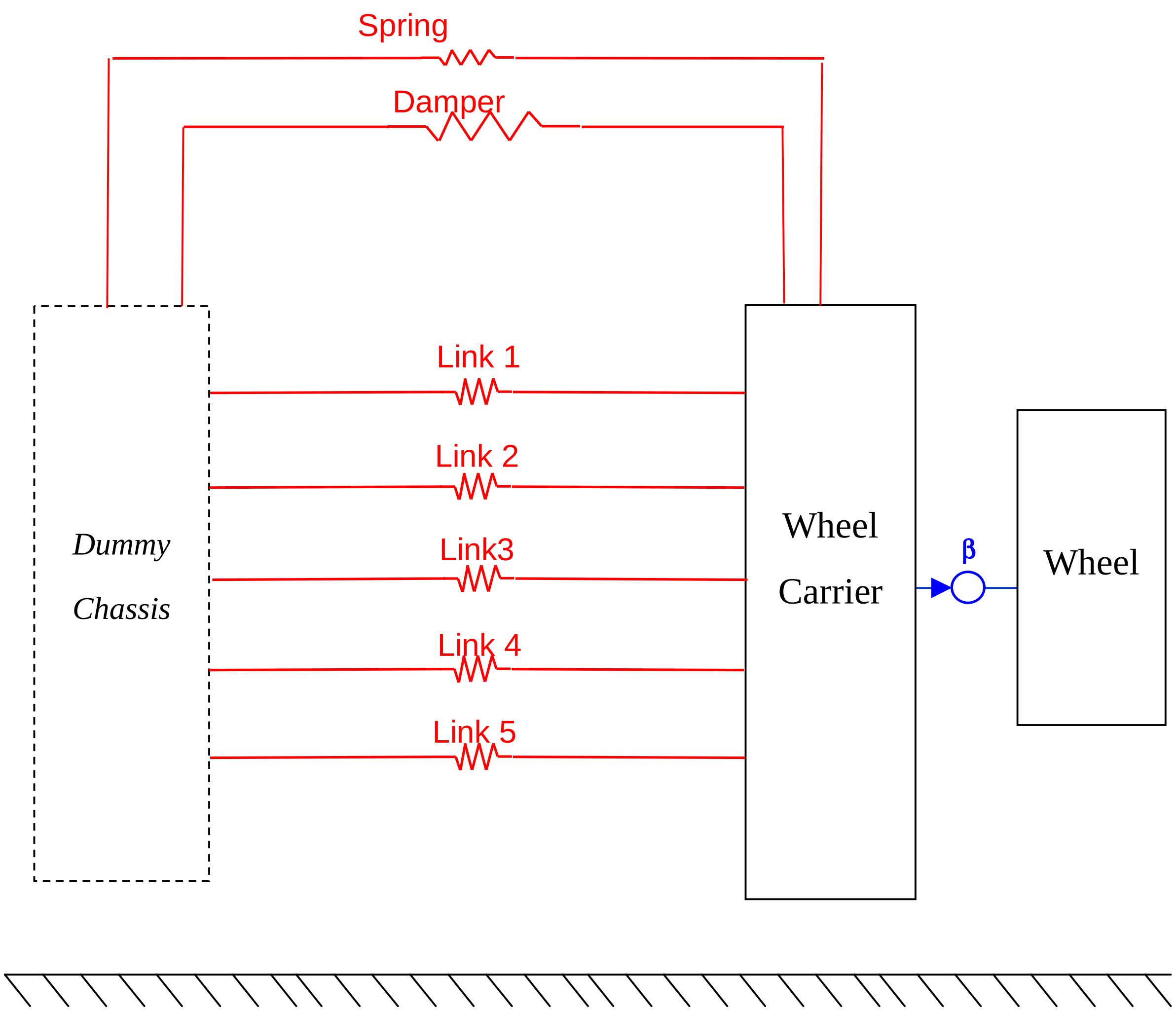


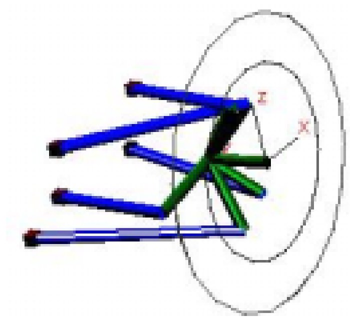


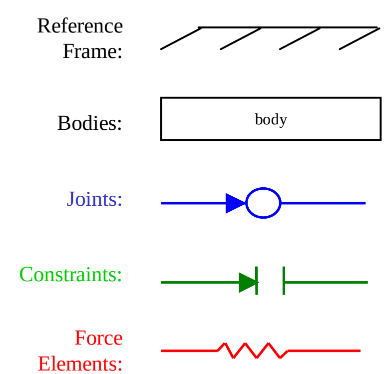


Application Example:Reduction of Rear Suspension

Reduced SIMPACK Model





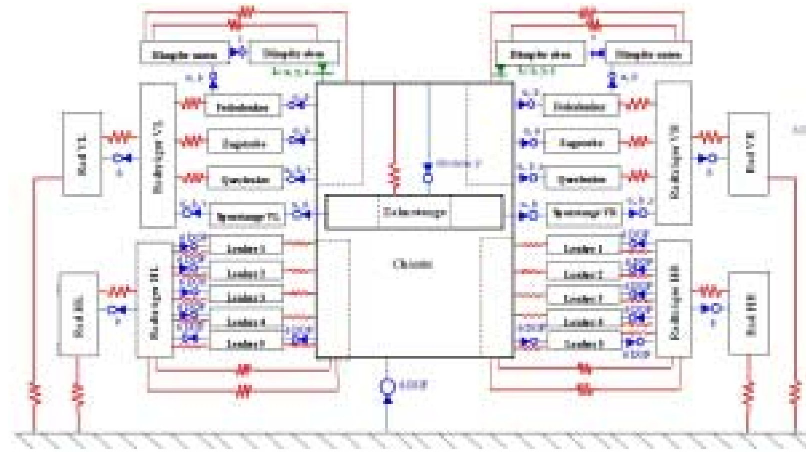


Application Example: Full Model vs. Reduced Model

Full Model:

number of bodies:	59
number of force elements:	60
number of 1. order states:	257
real-time factor with stiff, DAE-Solver ODASRT:	1:5

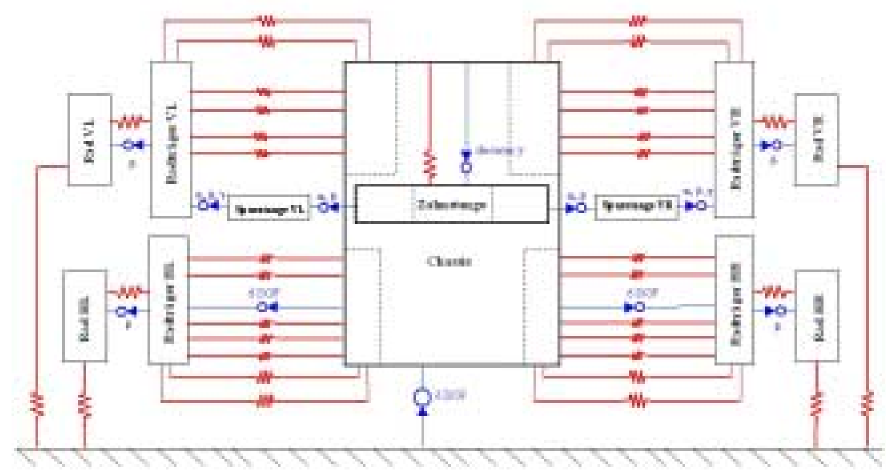
A



Reduced Model:

number of Bodies:	30
number of Force Elements:	54
number of 1. Order states:	64
CPU-time with expl. Euler 1,5 msec:	1:0,5
CPU-time with SIMPACK Code Export, expl. Euler 1,5 msec:	**1:0,2**

B



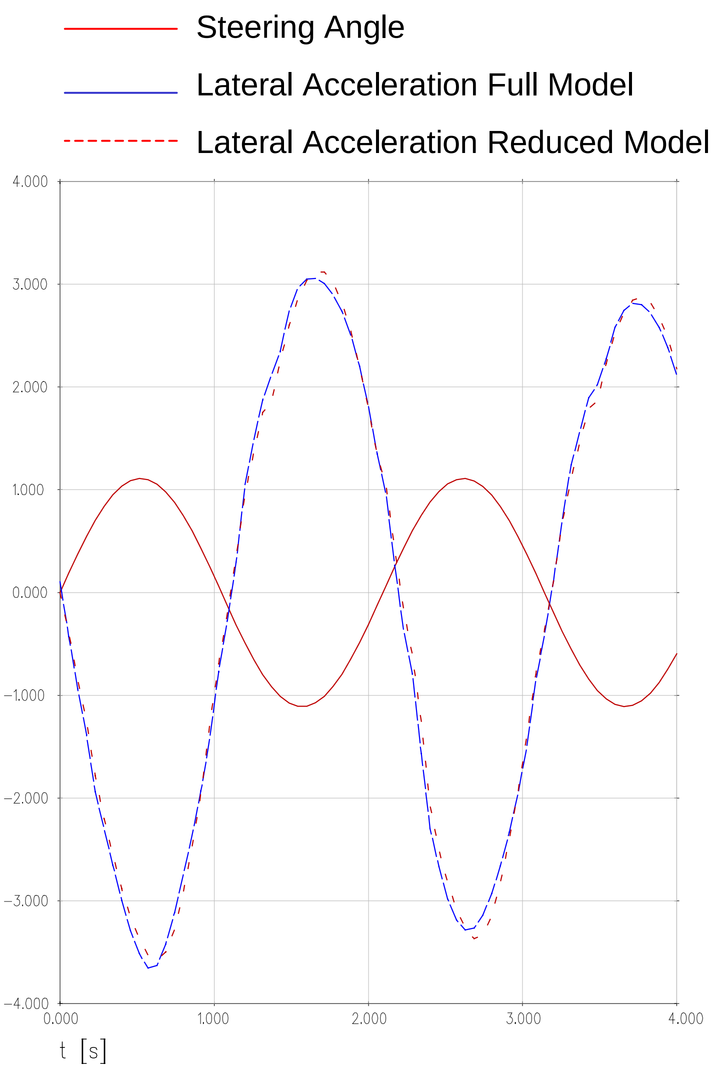
Summary

- ✓ Model reduction techniques resulted in a speed-up by factor 10
- ✓ **SIMPACK Symbolic Accelerator** added a factor 3 in speed-up resulting in a real-time model
- ✓ Additional speed-up could be gained by partial linearisation
- ✓ Account for flexible body components possible
- ✓ Automatic generation of real-time models from existing FVPs by **SIMPACK Code Export** ensuring one data base for off-line and real-time simulation

Open Fields

- Real-time solvers for numerically stiff systems and differential algebraic systems
- Identification environment for finding the parameters of a reduced real-time model
- Stable co-simulation of MBS and/or actuator models to support a multi-processor approach

Full vs. Reduced Model



Off-Line and On-Line: Different Worlds Today

Data Sources | **Converter** | **MBS Model**

Data Sources

FE-Components
e.g. Stabiliser Bar

CAD-Data
e.g. Geometry

Measurements
e.g.. Air Resistance
Suspension
Kinematics
Compliant Kin.
Tyres
...

Converter: For Off Line → **MBS Data Base** → MBS Model:

- **Kinematics**
- **Handling**
- **Stability / Derailment**
- **Ride Comfort**
- **Noise / Vibration / Harshness**
- **Durability / Fatigue**
- **Safety / Impact**

Converter: For On Line → **Real-Time Model Data** → **Real-Time Applications**

Off-Line and Real-Time: One World in Future

Data Sources | **Converter** | **MBS Model**

FE-Components

e.g. Stabiliser Bar

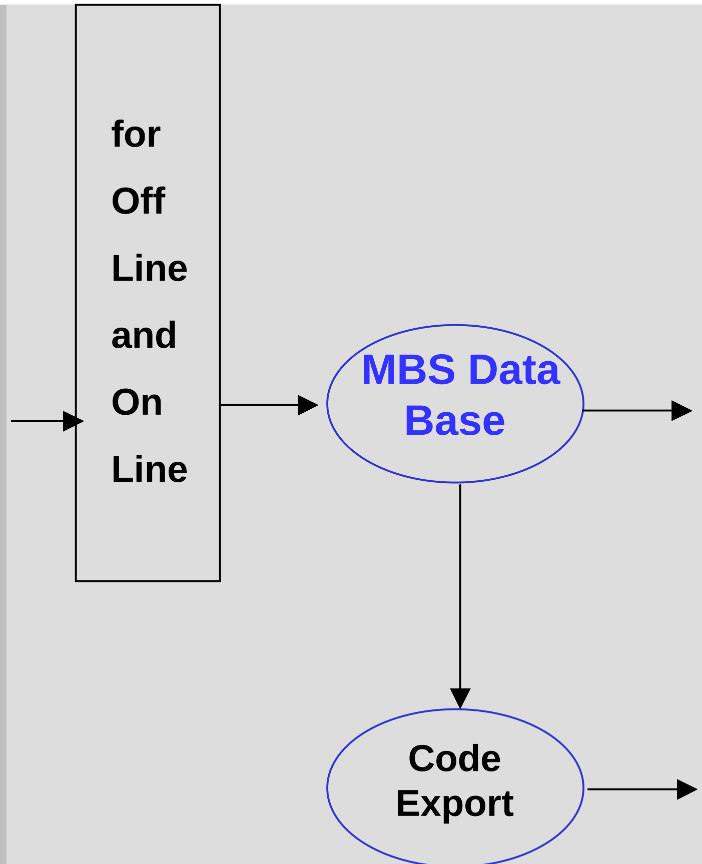
CAD-Data

e.g. Geometry

Measurements

e.g.. Air Resistance

- Suspension
- Kinematics
- Compliant Kin.
- Tyres
- ...



Kinematics

Handling

Stability / Derailment

Ride Comfort

Noise / Vibration / Harshness

Durability / Fatigue

Safety / Impact

Real-Time Model

Look-Up Tables for Suspension Kinematics