

Virtualisierung des Motorsteuergeräts basierend auf Object-File-Ansatz



Inhalt

1. Motivation

2. Umsetzung des Object-File-Ansatzes

3. Features des Object-Ansatzes

4. Anwendungsbeispiele

5. Fragen/ Diskussion

Inhalt

1. Motivation

2. Umsetzung des Object-File-Ansatzes

3. Features des Object-Ansatzes

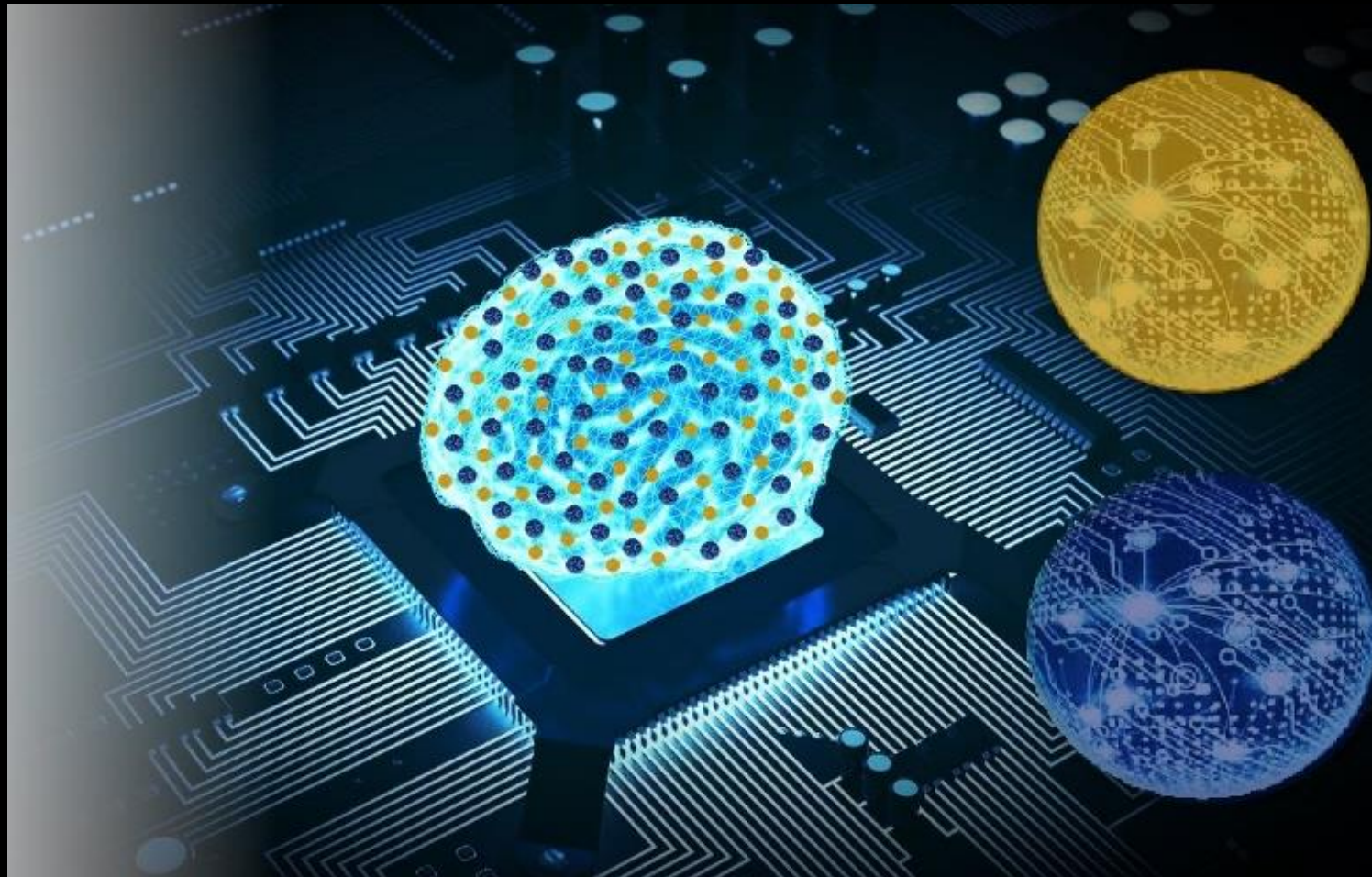
4. Anwendungsbeispiele

5. Fragen/ Diskussion

1. Motivation

- Die zunehmende Komplexität der Steuergeräte und der Anforderung an agile Entwicklung führen zu neuen digitalen Methoden
- Die Effizienz der Entwicklung steigt mit der „*Vollständigkeit*“ eines Steuergeräts

➔ Bedarf an Zulieferer-Anteil bei einem virtuellen Steuergerät



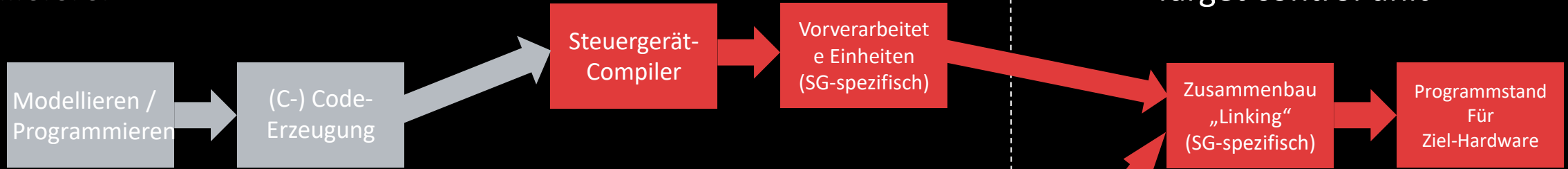
Inhouse-Software-Funktion

Zulieferer-Software-Funktion

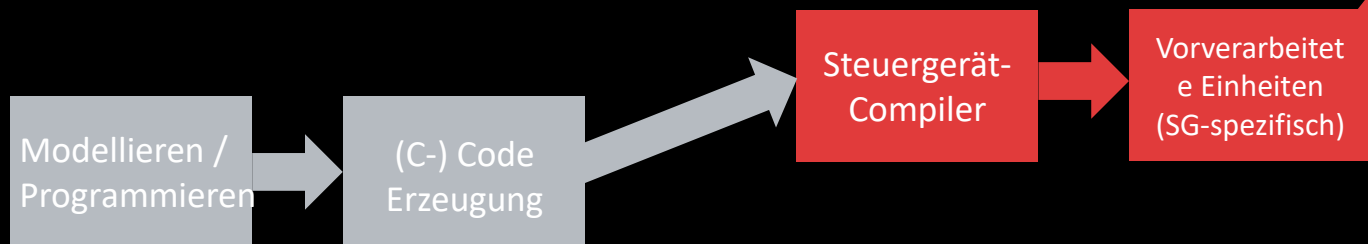
1. Motivation

Real-Target-Erstellungsprozess:

Zulieferer



Daimler / Inhouse



Wie ist es möglich vor dem Target-ECU ein virtuelles ECU zu generieren?

Inhalt

1. Motivation

2. Umsetzung des Object-File-Ansatzes

3. Features des Object-Ansatzes

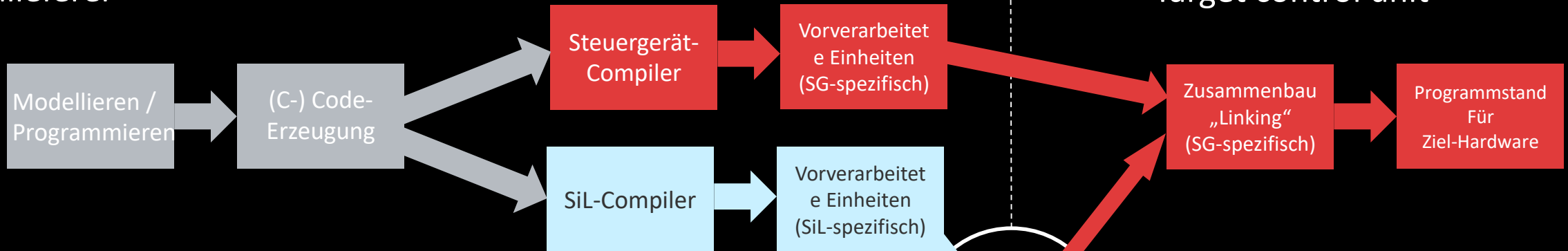
4. Anwendungsbeispiele

5. Fragen/ Diskussion

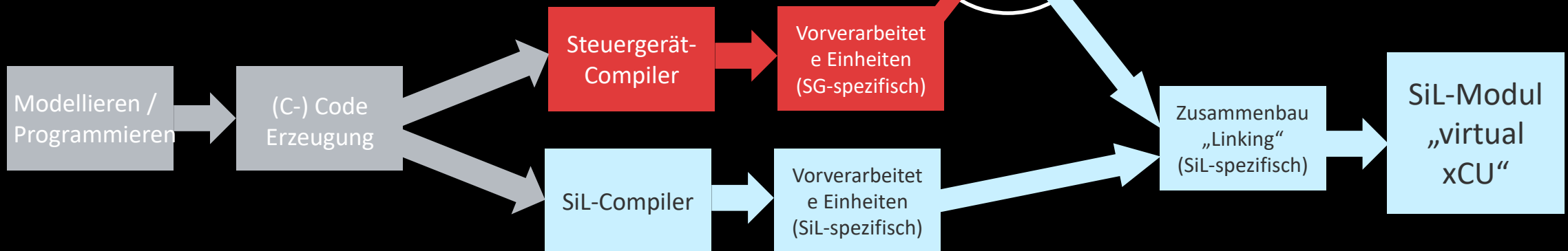
2. Umsetzung des Object-File-Ansatzes

- Virtualisierung-Prozess eines Steuergeräts nach **Object-File-Sharing-Ansatz**

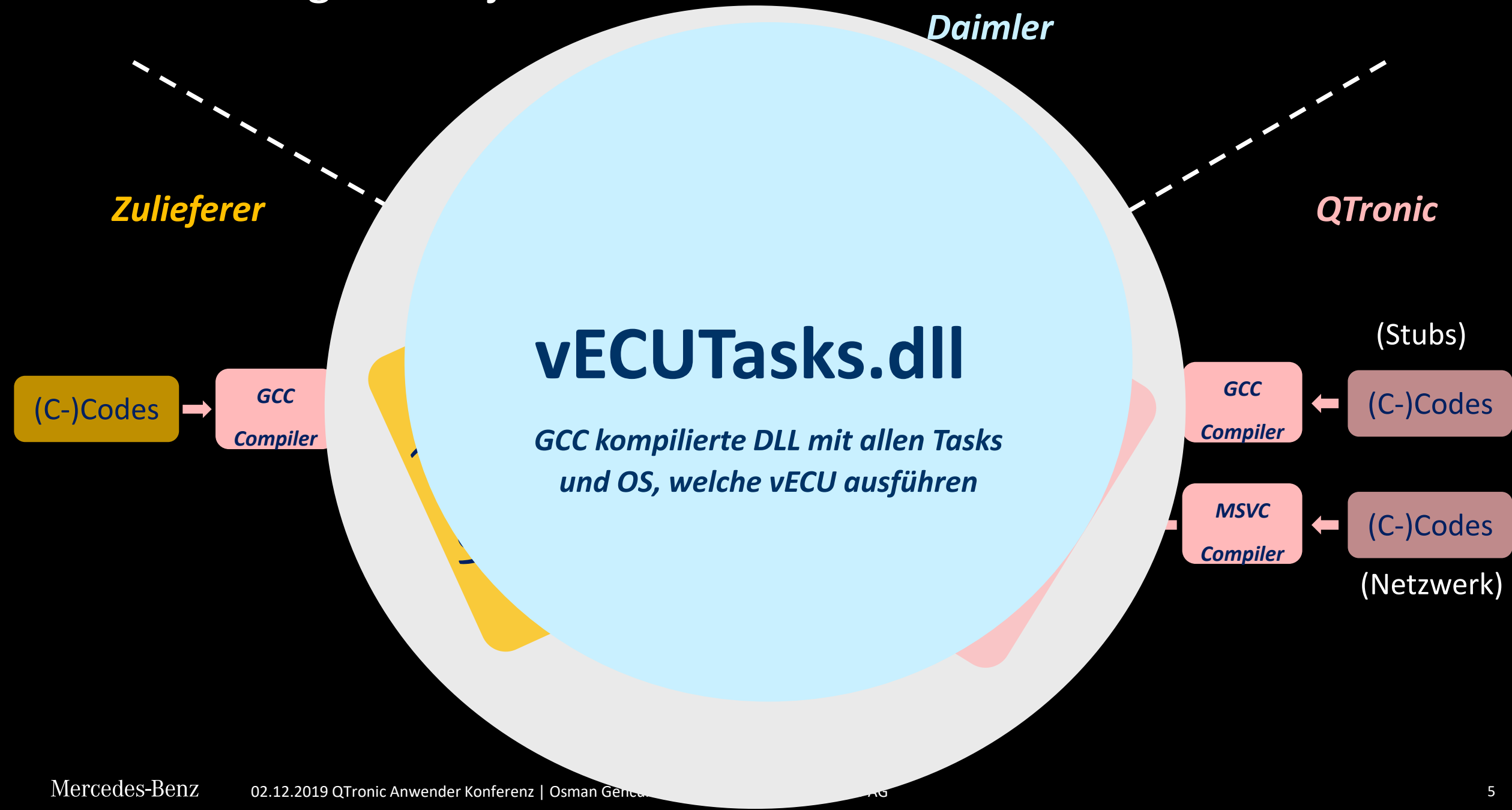
Zulieferer



Daimler / Inhouse



2. Umsetzung des Object-File-Ansatzes

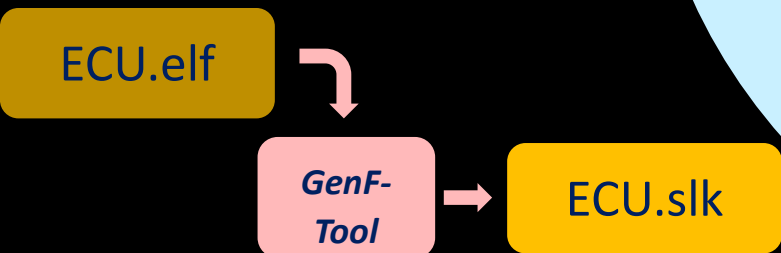
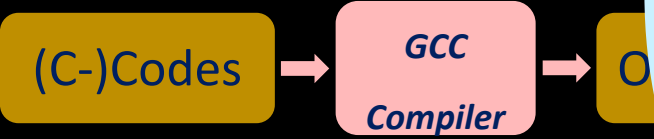
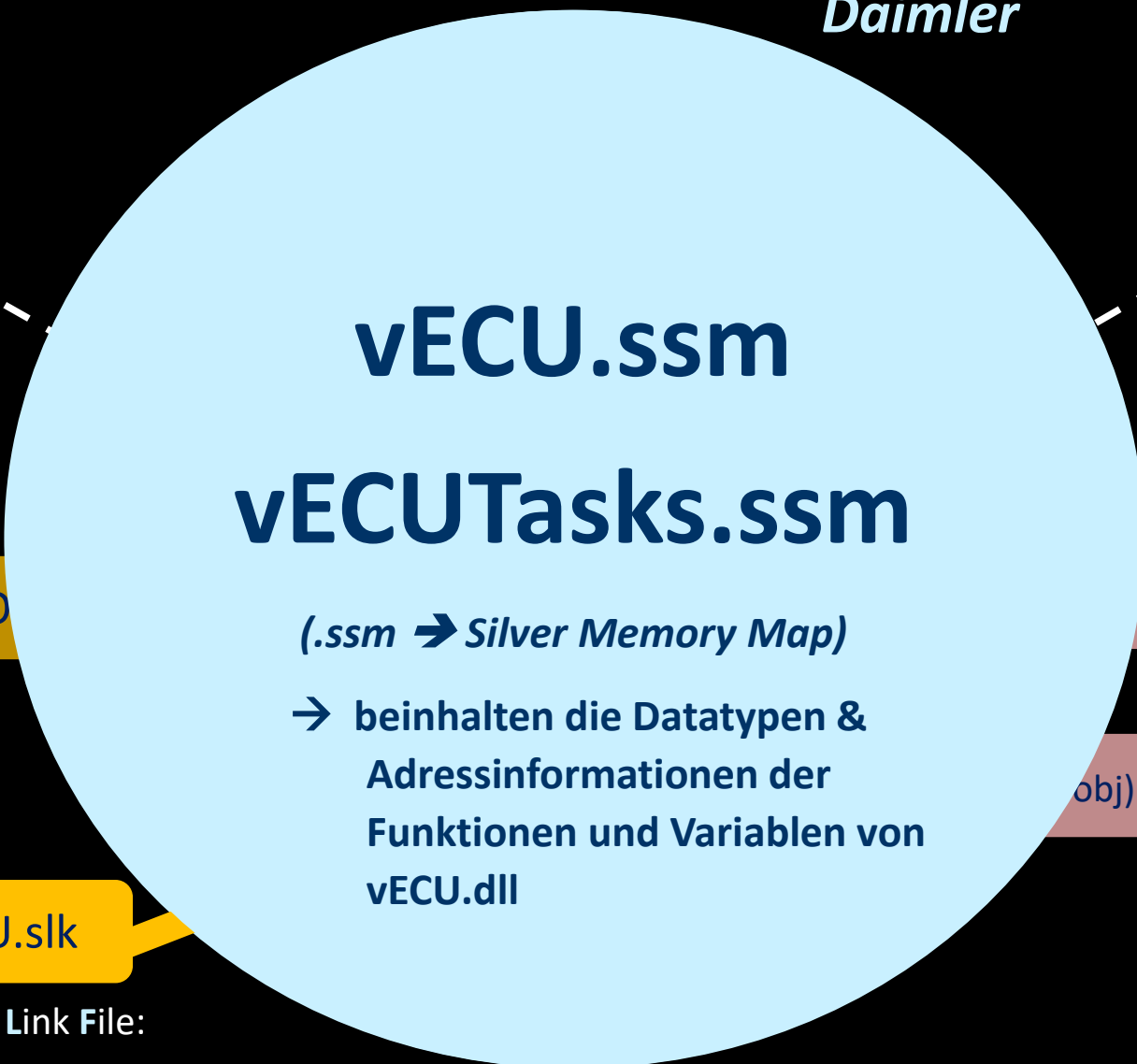


2. Umsetzung des Object-File-Ansatzes

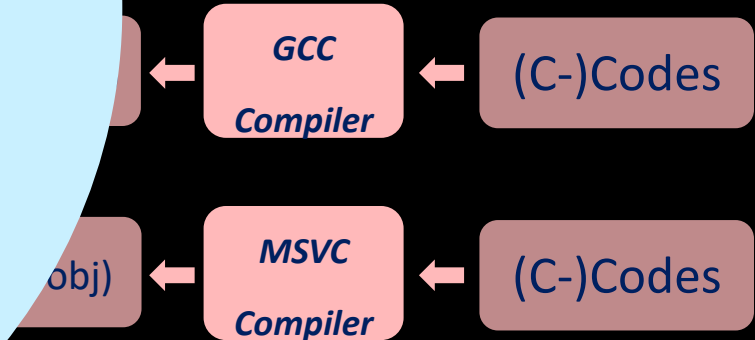
Daimler

Zulieferer

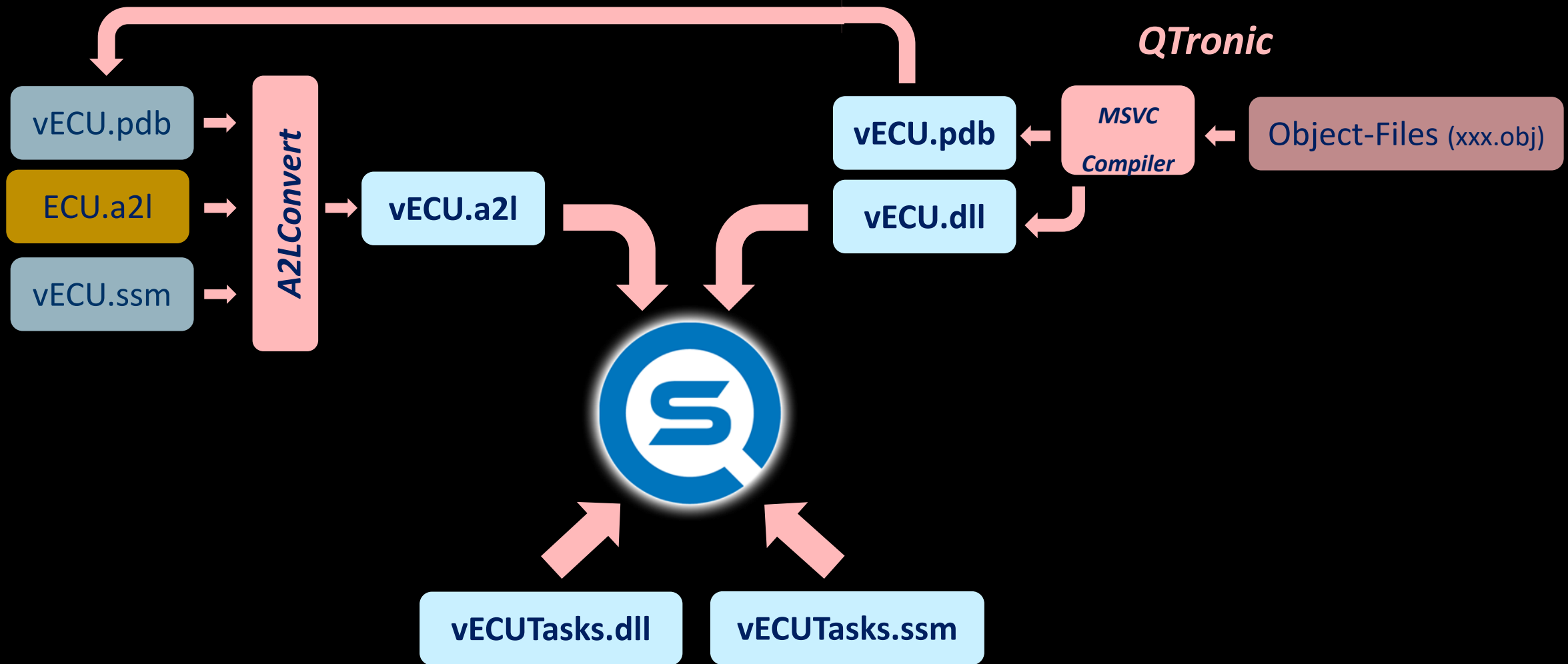
QTronic



Symbol Link File:
Zuordnung jedes A2L-Labels zum Symbol



2. Umsetzung des Object-File-Ansatzes



```
INFO: vECU.dll: Using vECUTasks.dll, which should contain all tasks and variables.
INFO: vECU.dll: Using additional map file vECUTasks.smm for name lookup
INFO: vECU.dll: Reading a2l-file vECU.a2l
```

Inhalt

1. Motivation

2. Umsetzung des Object-File-Ansatzes

3. Features des Object-Ansatzes

4. Anwendungsbeispiele

5. Fragen/ Diskussion

3. Features des Object-Ansatzes

Integration von BSW ermöglicht den Schnitt an:

- Complex Driver
 - Die nötigen Hardware-Teile sind ebenfalls emuliert (z.B. 60-2 Geberrad)
- ASICS-Emulation
- Kommunikation

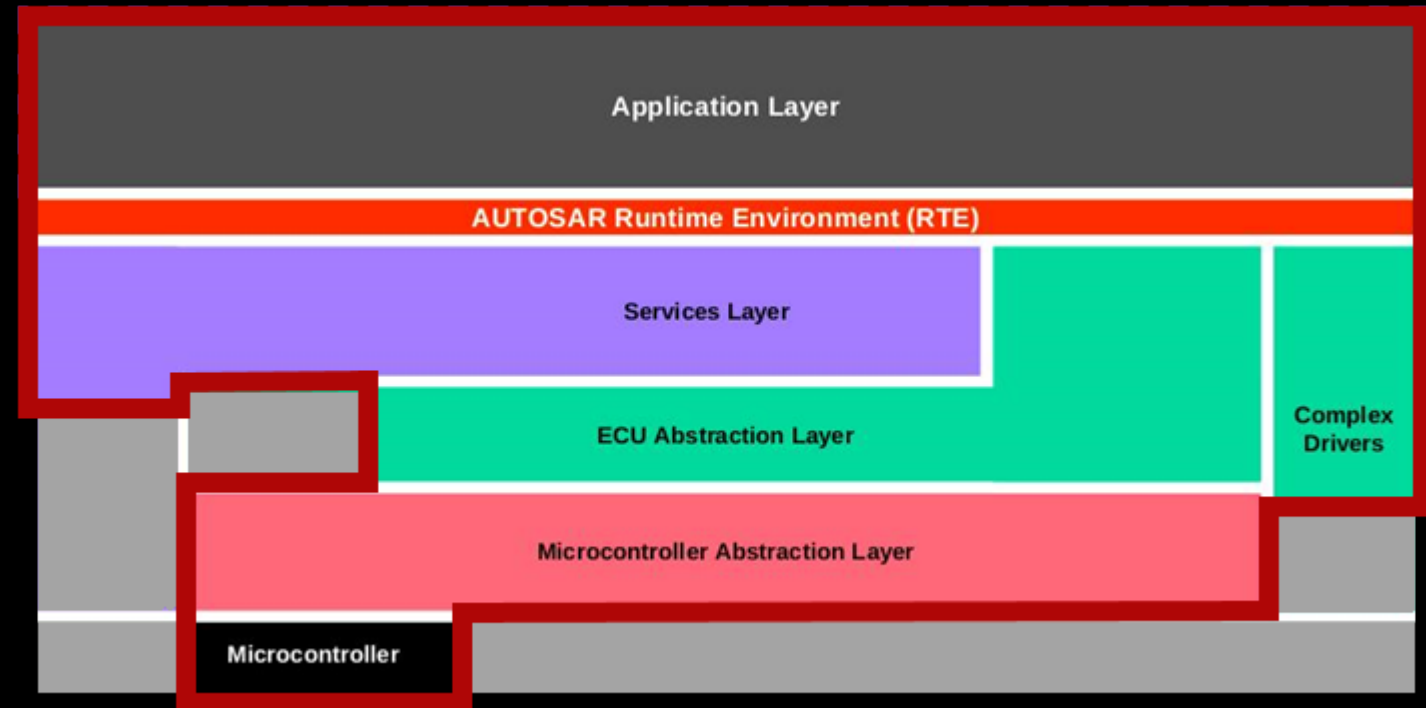
Interface mit Simulationssystem

- Zugriff auf Applikationsgrößen
- Zugriff auf Hardware Ein- /Ausgänge
- Integration der originalen OS-Scheduler

Unterschiedliches Target

- Anderes Verhalten ECU ↔ vECU (div0)
- Prozessor-spezifischer Code

Schnitt ohne BSW beinhaltet nur ASW



Inhalt

1. Motivation

2. Umsetzung des Object-File-Ansatzes

3. Features des Object-Ansatzes

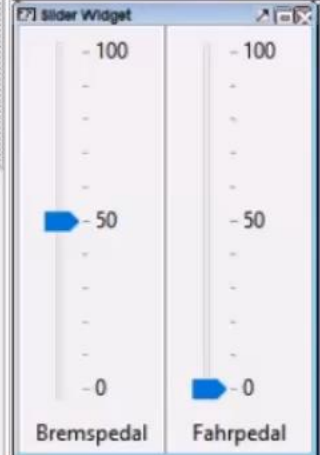
4. Anwendungsbeispiele

5. Fragen/ Diskussion



Label Widget

ZÖP	
TxHybDrvMd_Rq_ECM_V2	0 PT_DISENGG
TrsmCtrl_HybDrvMd	0 PT_DISENGG
HDC_HybDrv_Stat	7.000
Gr	P
Gr_Target	P
PT_Rdy_V2	
PT4_VEPM_Energy_Rq_ST3	0
SPCR_Sp_Stat	IDLE
PT4_PTCoor_HDC_HybSync_Stat_ST3	0.000
PT4_PTCoor_HDC_HybSys_Stat_ST3	0.000



Canvas Widget

Max-Range	<input type="radio"/>	EngTrq	0
Comfort	<input checked="" type="radio"/>	EM01_Trq	0
Eco	<input type="radio"/>	EngTrqStatic	0

5. Fragen/ Diskussion



Mercedes-Benz

Das Beste oder nichts.

