

RACE Praktikum Student Competition

Which functional features do you expect from your future electric vehicle?

Why not implement them yourself? ...do it now!

21.01.2014

Website:

<http://www4.in.tum.de/lehre/praktika/map/ss14/raceapps>

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages



Project RACE

Robust and reliant Automotive Computing Environment for future eCars



Gefördert durch:



Bundesministerium
für Wirtschaft
und Energie

- Funded by BMWi
- Project time: January 2012 – December 2014

<http://www.projekt-race.de>

- Project based on study „Mehr Software (im) Wagen“

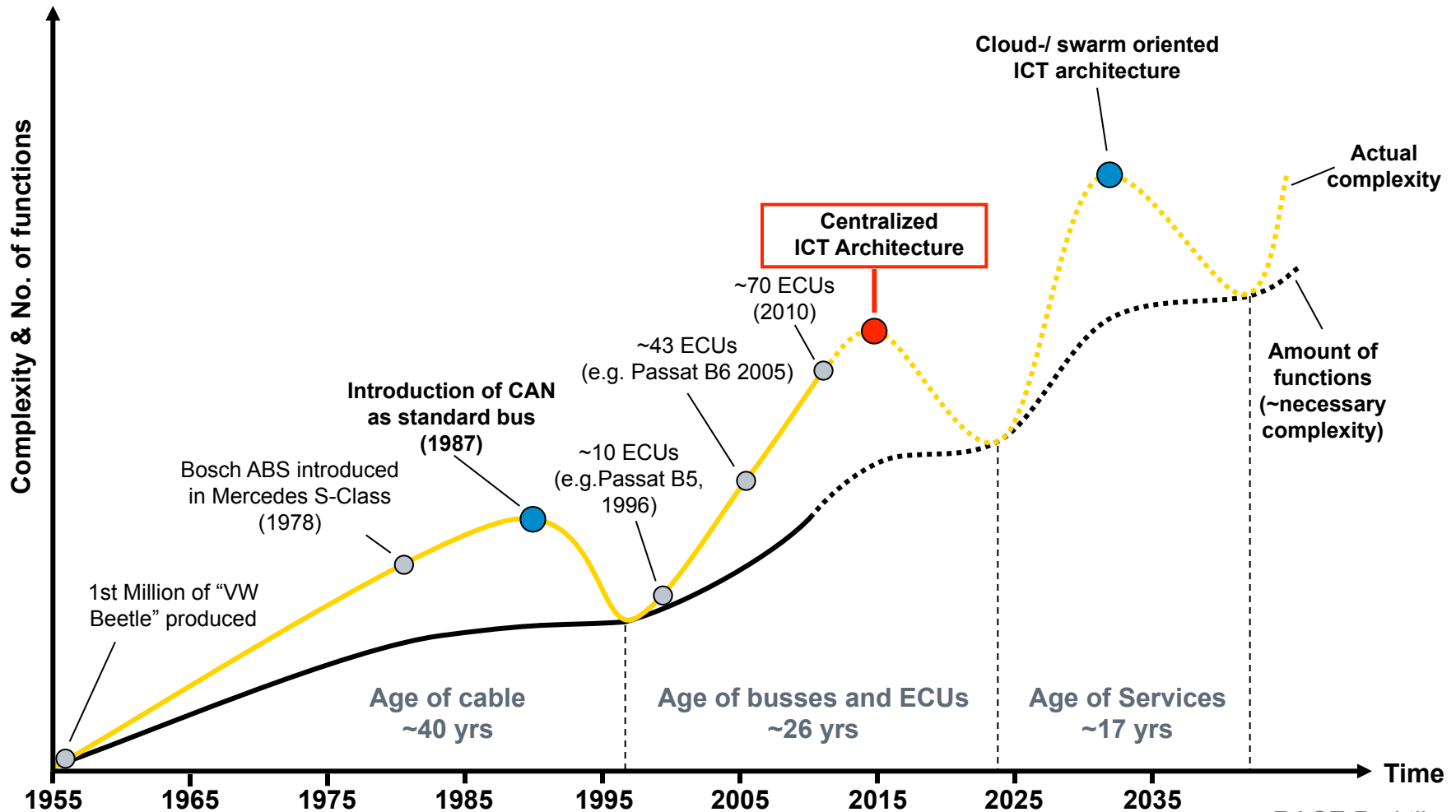
<http://www.fortiss.org/ikt2030>



aufgrund eines Beschlusses
des Deutschen Bundestages



History and forecast about the complexity of automotive architectures



Source: "The Software Car: Information and Communication Technology as an Engine for the Electromobility of the Future", page 48

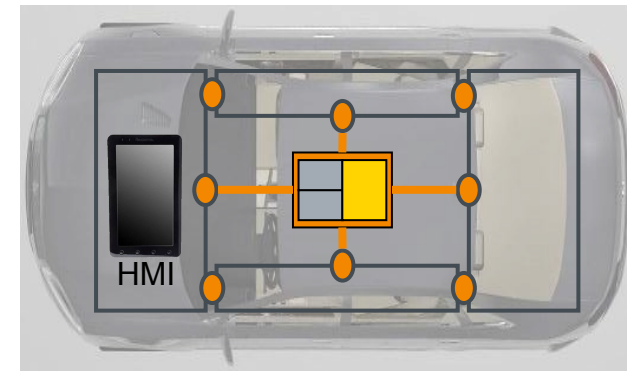
RACE Platform Idea

Main Project objectives:

- **Aim 1:** Reduction of complexity of ICT-Architecture by homogeneous and open basis platform
- **Aim 2:** Support if new complex functional vehicle features
- **Aim 3:** Plug & Play capability of ICT-Architecture
- **Aim 4:** Ability to certify the ICT-Architecture
- **Aim 5:** Show an migration path to the new architecture

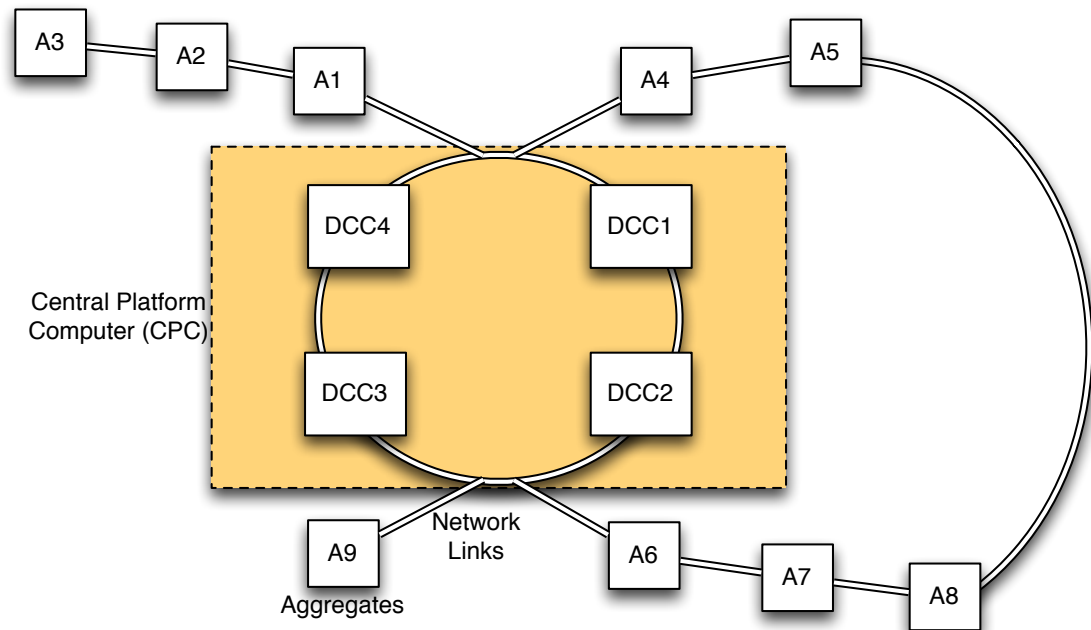
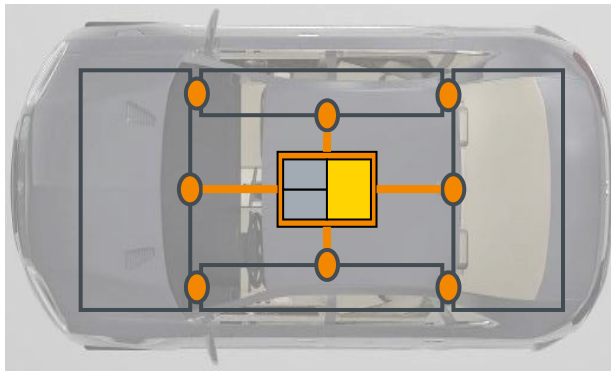
Main Principles:

- Centralized ICT-Architecture
 - Central-Platform-Computer, mixed critical features
 - Data-centric approach: All data about Sensors and Actuators is accessible everywhere
- Communication
 - Switched Ethernet
 - Publish/Subscribe communication pattern
- Fail-Operational vehicle features



RACE Platform

- Central Platform Computer (CPC) with scalable set of Duplex-Control-Computers (DCCs) and safety concepts to enable fail-operationality
- Data-Centrism allows freedom on deployment
 - Application-Components can access Sensor/Actuator data independently from the components location

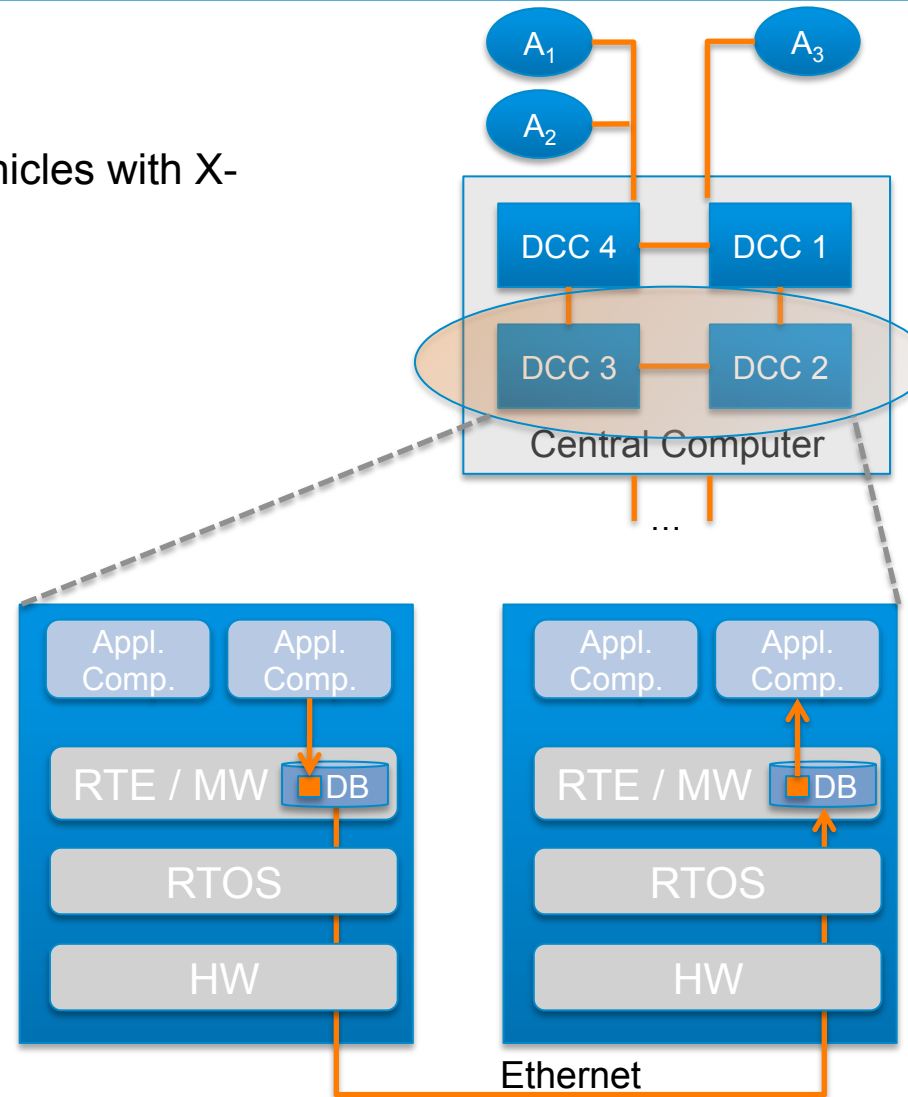


RACE Platform

- Centralized ICT-Architecture for electric vehicles with X-By-Wire support
- Cluster of central computing units
- Data-Centric approach
 - all data accessible at every location
- Logical execution times

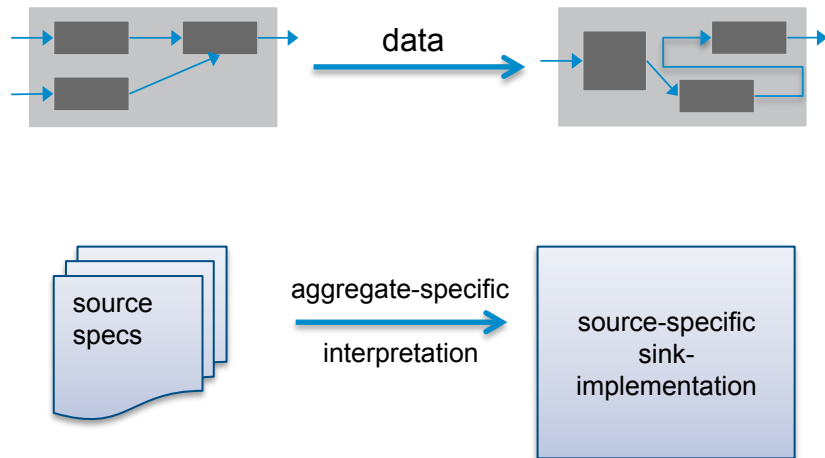
Main Requirements:

- Mixed-Criticality
- Fault-Detection & Handling
- Fail-Operationality



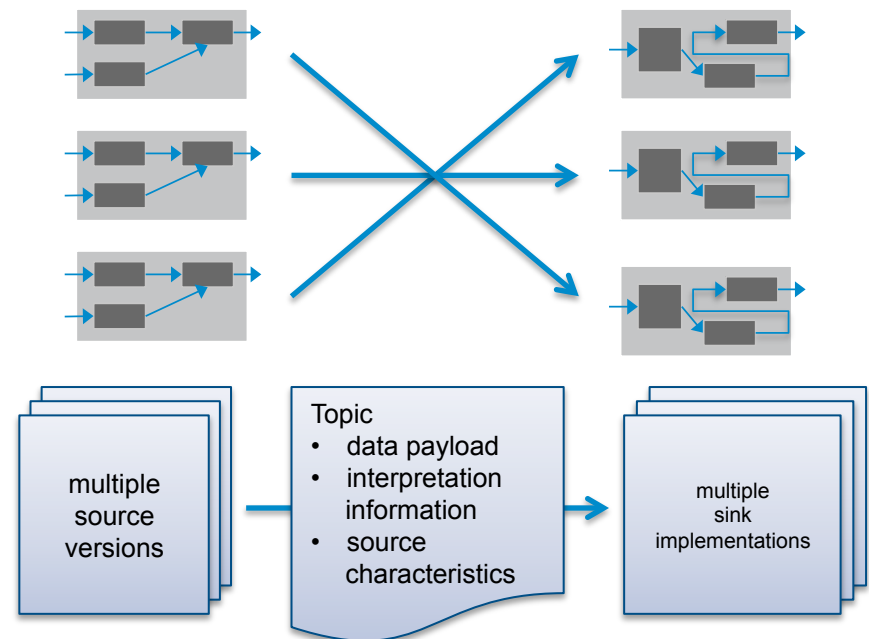
Data-Centric Communication

- Comparison of Approaches



Classic Approach: Aggregate-Centric

- Data is interpreted using knowledge about the source
- Dependent on (informal) knowledge
- Tight coupling of aggregates and aggregate developers needed
- Inflexible, hard to build modular architectures

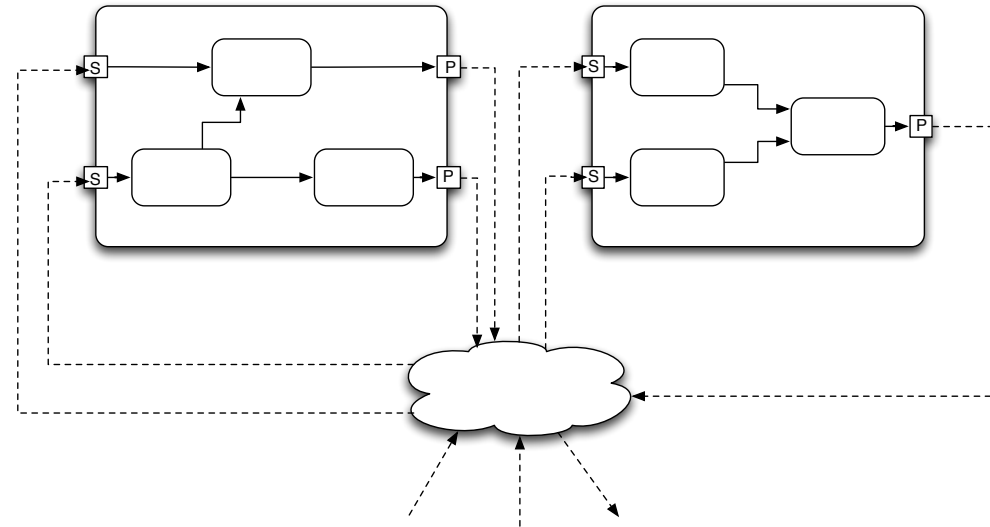


Novel Approach: Data-Centric

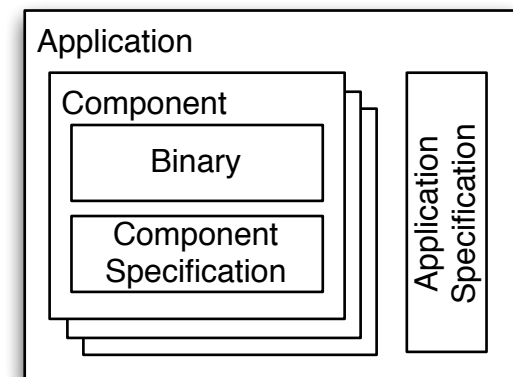
- The source properties are reflected in the topic **attributes**
- Data can be interpreted solely relying on the topic description
- **Decoupling** of producer and consumer of data
- **Flexible**, enables **modular** architectures

RACE Application Development Fundamentals

- Applications are realized by black-box SW-Components
- External interfaces of the SWCs are specified by Ports that subscribe or publish data.
- Internal sub-components are hard wired.



- Applications are delivered together with specifications (so called *Manifests*) about their functional and non-functional properties and requirements
 - e.g. WCETs, required memory, required and provided data, etc.
 - These information is required to decide about the integrability/composability of the application components



Demonstrator Car

In the RACE Project, 3 vehicle functions get implemented to evaluate the developed vehicle platform:

1. Energy-Management
2. Driving (trajectory-based)
3. Autonomous parking at a charging station

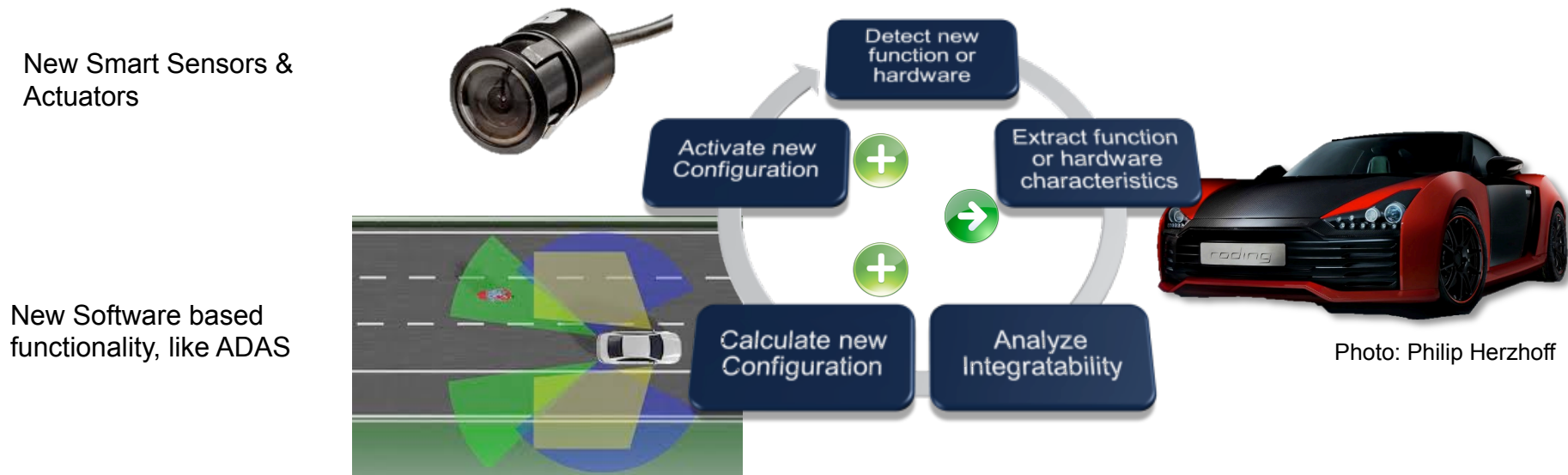


Photo: Philip Herzhoff

During the RACE-Praktikum, additional functions should be invented and implemented. The best implementations might also get integrated into the demonstrator car.

Plug and Play of new functionality

Evolutionary update of the vehicle through autonomous integration of new software and hardware components, enabling new functionality



Organization

Phase 1 Idea Contest

- Brainstorming and conception of innovative future vehicle functions, based on the vehicle platform developed in RACE

Phase 2 Implementation

- Part 1: Implement functionality on central platform computer
- Part 2: Implement visualization and control of functionality on HMI.

Phase 3 Simulation, Evaluation, Documentation and Integration

- Simulation and Evaluation of implemented functionality
- Creation of work report
- Optional: Integration into real demonstrator car (via Plug-and-Play).

Phase 4 Presentation

- The best ideas and implementations of innovative functions can be presented at the final demonstration of the RACE project to a broad audience (Nov 14)

Registration

- February 2014

Phase 1:

- 1st half of SS14
- Wednesday 9-11h

Phase 2+3:

- 2nd half of SS14 and/or as block 2-3 weeks in lecture-free time in August/Sept.
- You can decide.

Teamwork

- 3-4 Students per Team

TUM-Praktikum

- 10 ECTS, 6 SWS

Questions? Contact: Klaus Becker (becker@fortiss.org)

Prerequisites

Phase 1 Idea Contest

Prerequisites for Phase 1:

- Basic knowledge of embedded systems, like in the automotive domain.
- You will get documentation about available sensor-data of the vehicle from us.
- Enthusiasm for vehicles and software ;)

Phase 2 Implementation on central platform computer and HMI

Prerequisites for Phase 2 and 3:

- Basic knowledge in **C** programming language and IDE **Eclipse** is assumed.
- Knowledge in **HTML5** and **JavaScript** is helpful for implementing the HMI GUI
- Experience with Matlab/Simulink might be helpful, but not required (optional)

Phase 3 Simulation, Evaluation, Documentation and Integration

Phase 4 Presentation

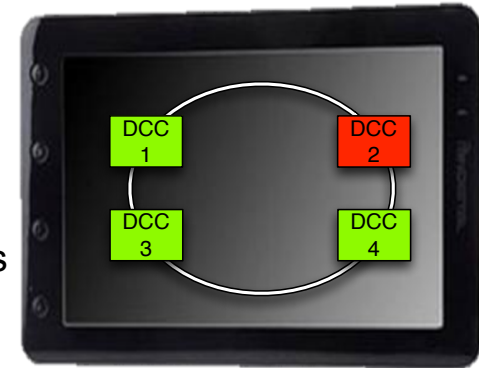
Not every student needs knowledge in every sector! => Teams!

Questions? Contact: Klaus Becker (becker@fortiss.org)

Just some ideas for functions

Visualization of the structure and status of the Platform:

- Visualization of states of hardware components
- Visualization of states of software components
- Visualization of deployment of software components to hardware components



Integration of external web-services

Provide higher services based on vehicle data

Active or passive “Advanced Driver-Assistance Systems” (ADAS)

etc.

Questions? Contact: Klaus Becker (becker@fortiss.org)

Idea for an „innovative“ HMI-App that uses passive vehicle data and generates a higher Service from this

Student Project VEHO:

<http://www.vehodrive.com/>

The Center for Digital Technology and Management (CDTM) developed "VEHO" - your perfect copilot. VEHO works with data out of the car, gathered by the OBD-device, and matches the data with external data like speed limits and weather information. This matching process results in real-time recommendations for the driver, improving his driving skills and presenting it in an overall score and statistics of the driving performance.



Done by 5 Students in WS12/13
at CDTM (<http://www.cdtm.de>).

Vielen Dank für Eure Aufmerksamkeit.

- Fragen ?

Website:

<http://www4.in.tum.de/lehre/praktika/map/ss14/raceapps>

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

