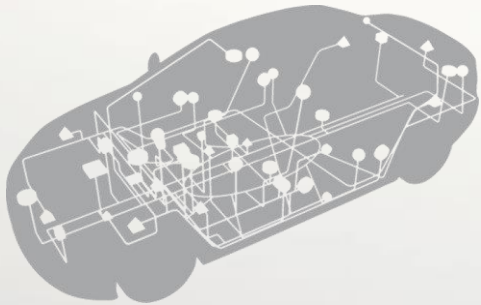


# Vehicle Application Plugin Platform



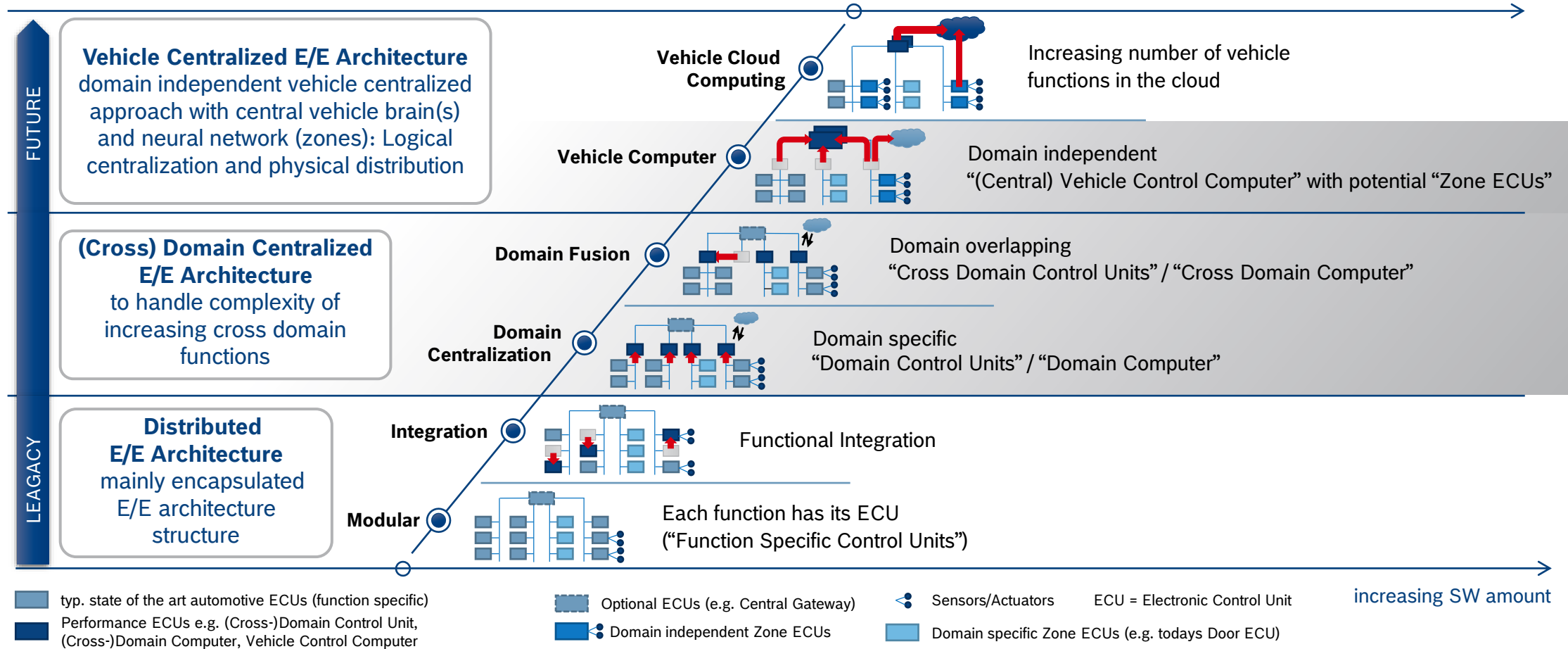
## Common Vehicle Interface Initiative

Robert Bosch GmbH  
M/NET      Dr. Christian Kerstan



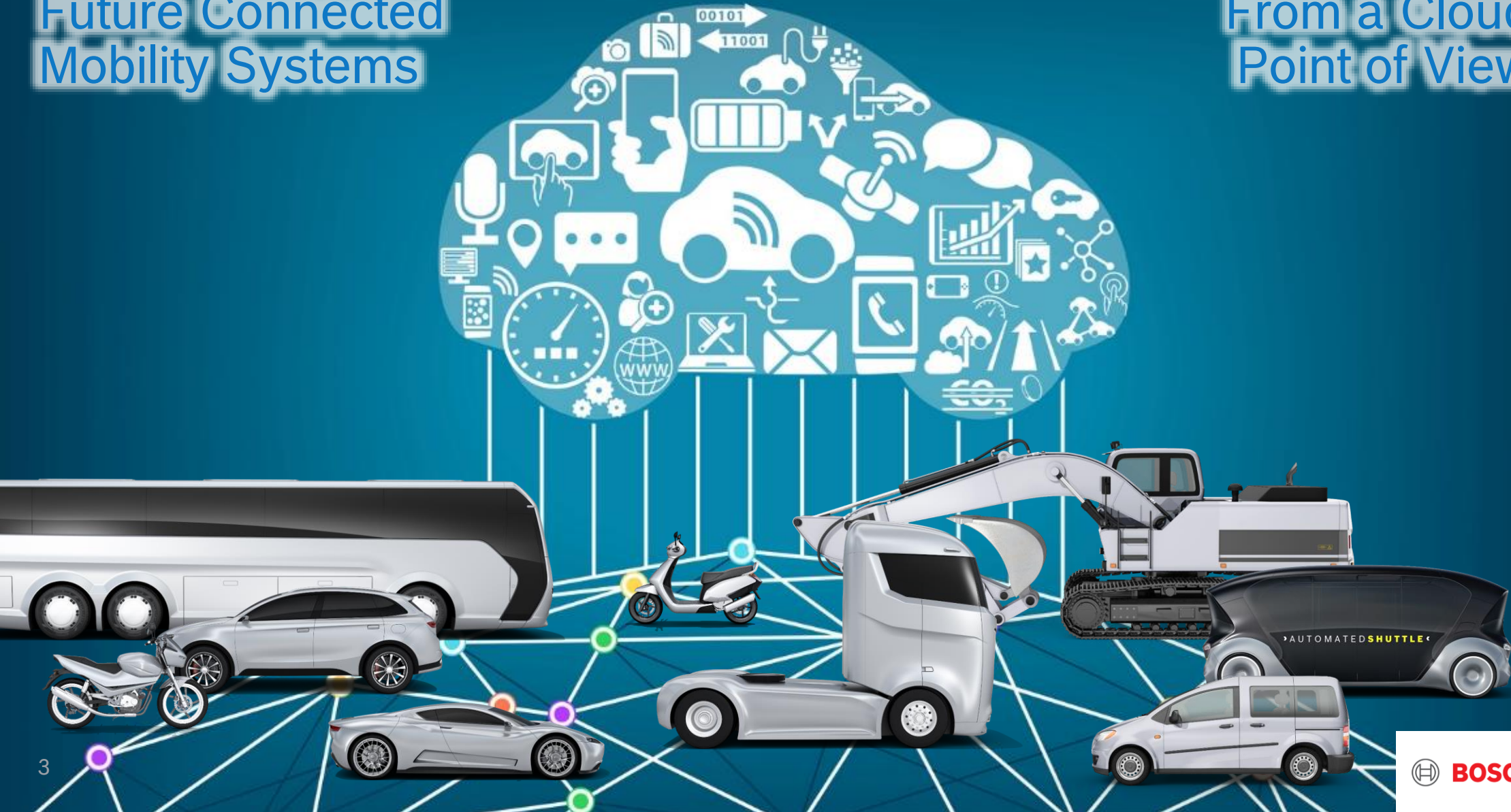
# Trends for Future Mobility Systems

## E/E Architecture Roadmap

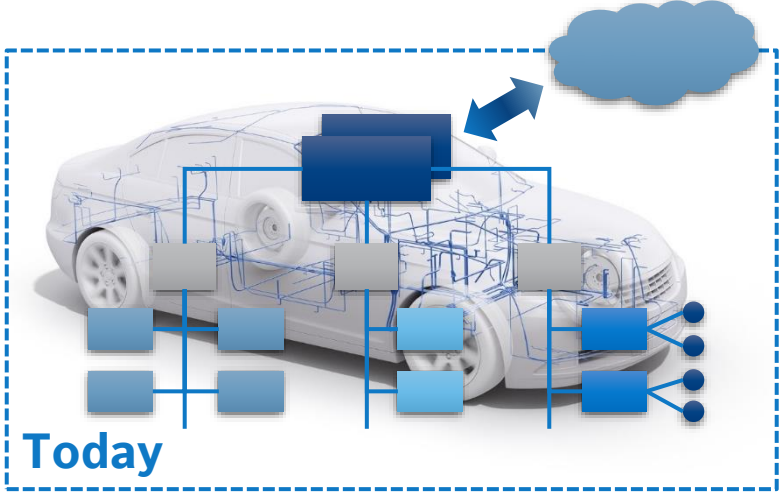


# Future Connected Mobility Systems

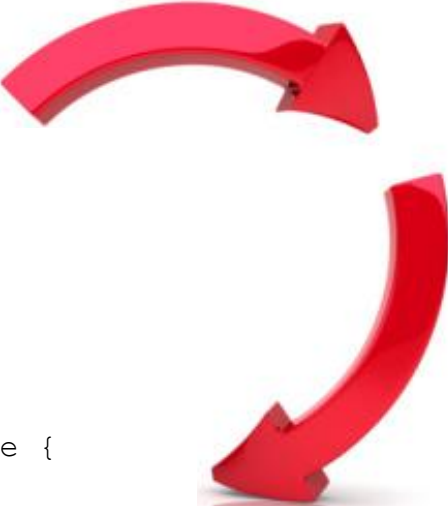
# From a Cloud Point of View



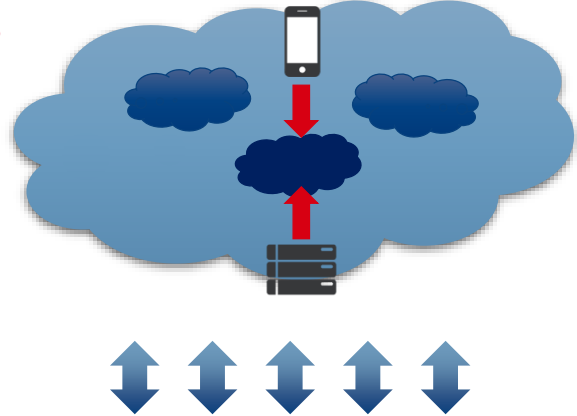
# Vehicle Application Plugin Platform “Rework” Functional Architecture



```
// dedicated vehicle implementation
class OEM_A_Mod_B_Var_42 : public Vehicle {
public:
    int getCurrentSpeed();
    ...
};
```



~~Future~~  
Now!

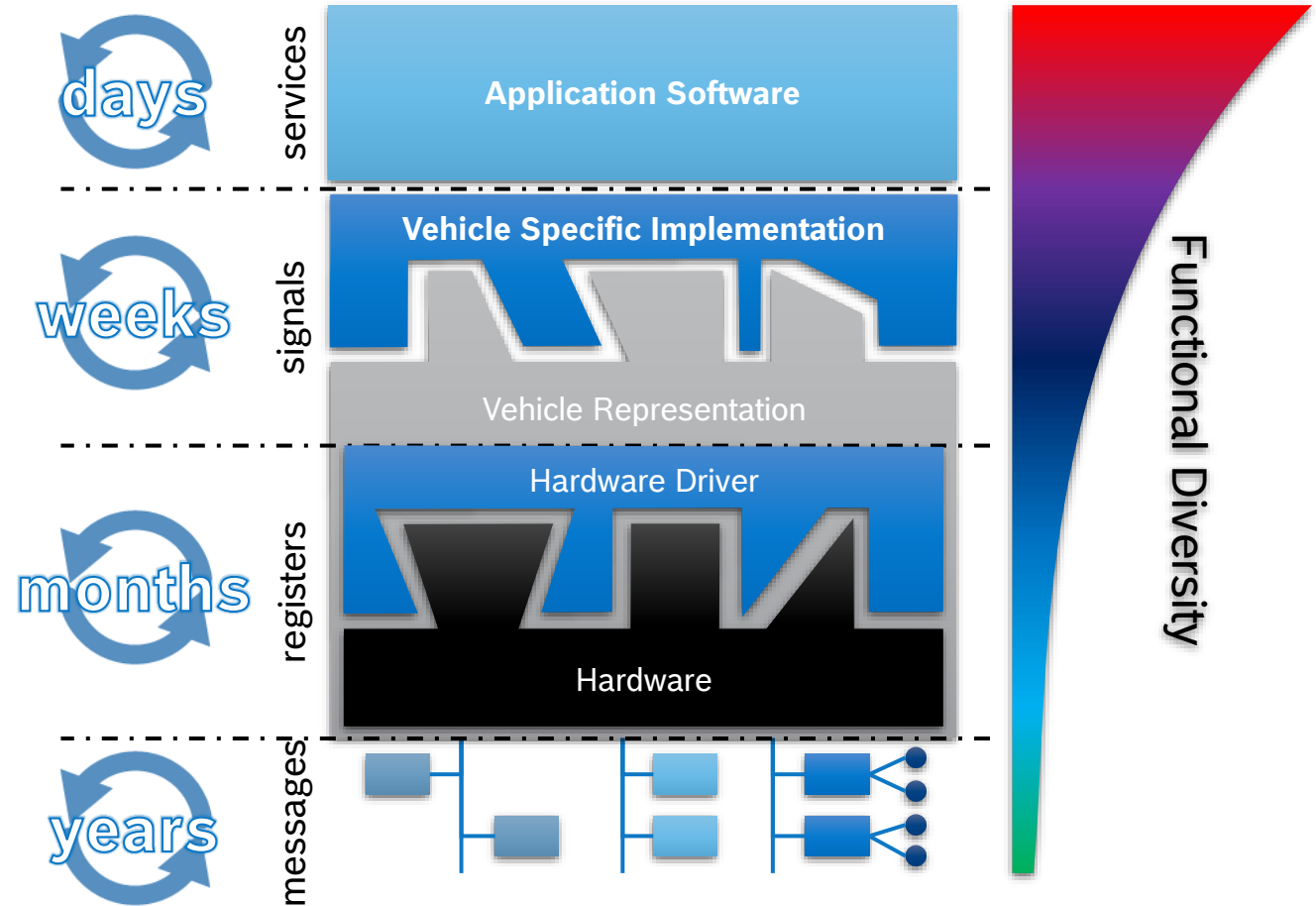


```
class Vehicle {
public:
    virtual int getCurrentSpeed() = 0;
    virtual void setSpeed(int speed) = 0;
    virtual bool getLock() = 0;
    ...
};
```

# Vehicle Application Plugin Platform

## Decoupling of Development & Deployment Cycles

- ▶ Decoupling of implementation reduces effort and complexity
- ▶ Decoupling of deployment cycles allows fast updates for high level features and well-proven processes for embedded functionality
- ▶ Service development does not require knowledge of all future functionality
- ▶ New business models possible due to independent deployment

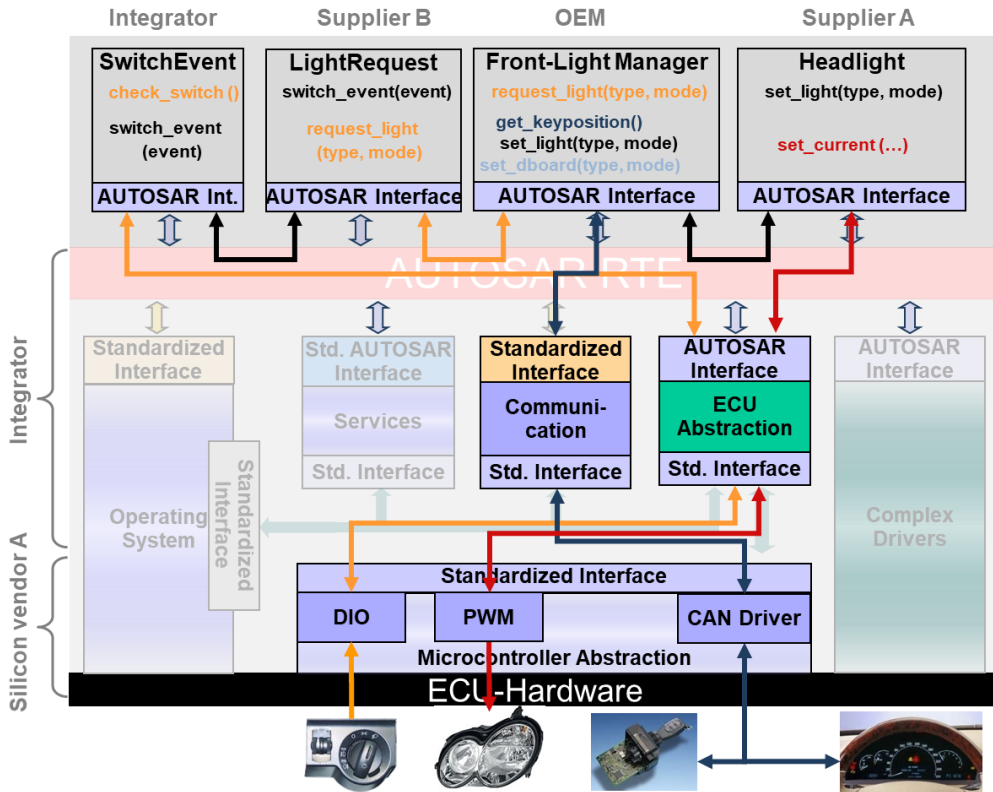


# Vehicle Application Plugin Platform

## Example: AUTOSAR: Exchange type of Front Light

[1/2]

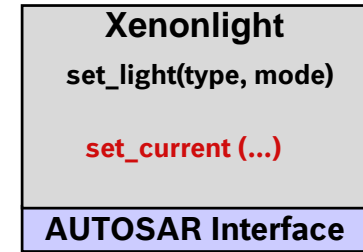
### Halogen Head Light



Exchange SWC



### Xenon Head Light



Reconfigure BSW



Exchange Hardware

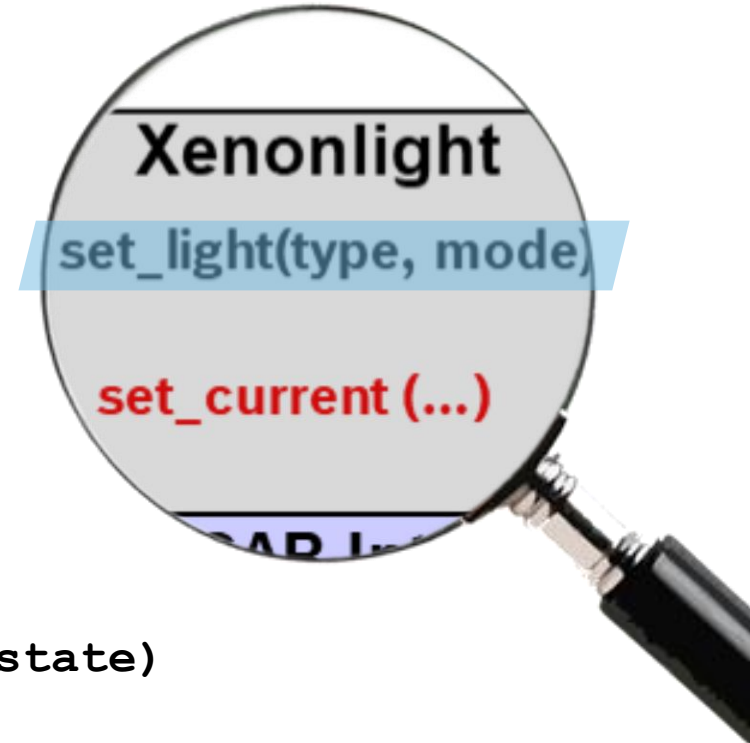


# Vehicle Application Plugin Platform

## Example: AUTOSAR: Exchange type of Front Light

[2/2]

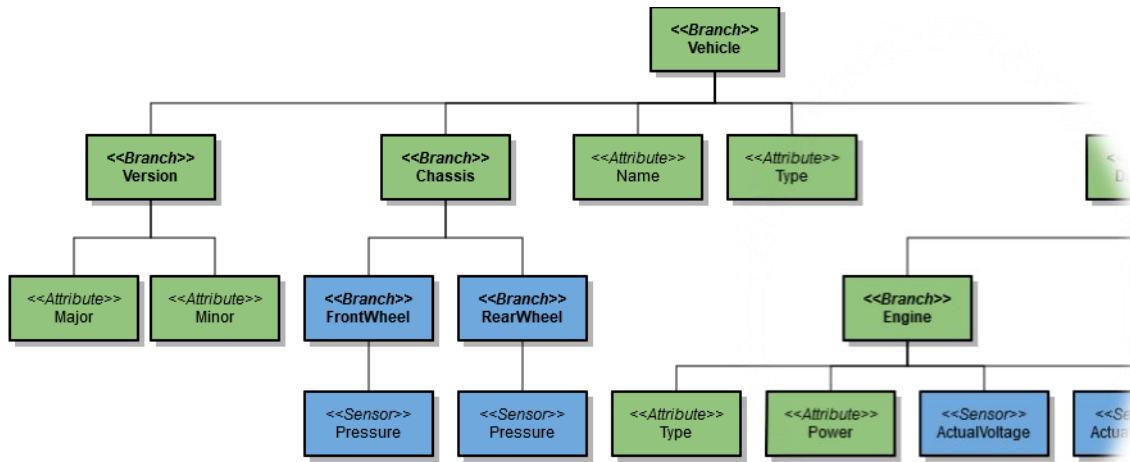
```
Set_Light(bool state)           setLight(enum state)
                                switchHeadLight(enum type, enum mode)
lightOn()                       setLight(bool state)
                                SetLight(bool state)
                                Set_Beam(enum range)
OP_MOD_Light_Func2(enum param1) g_DrvReqHB(enum state)
lightSwitchEvent(enum state)
```



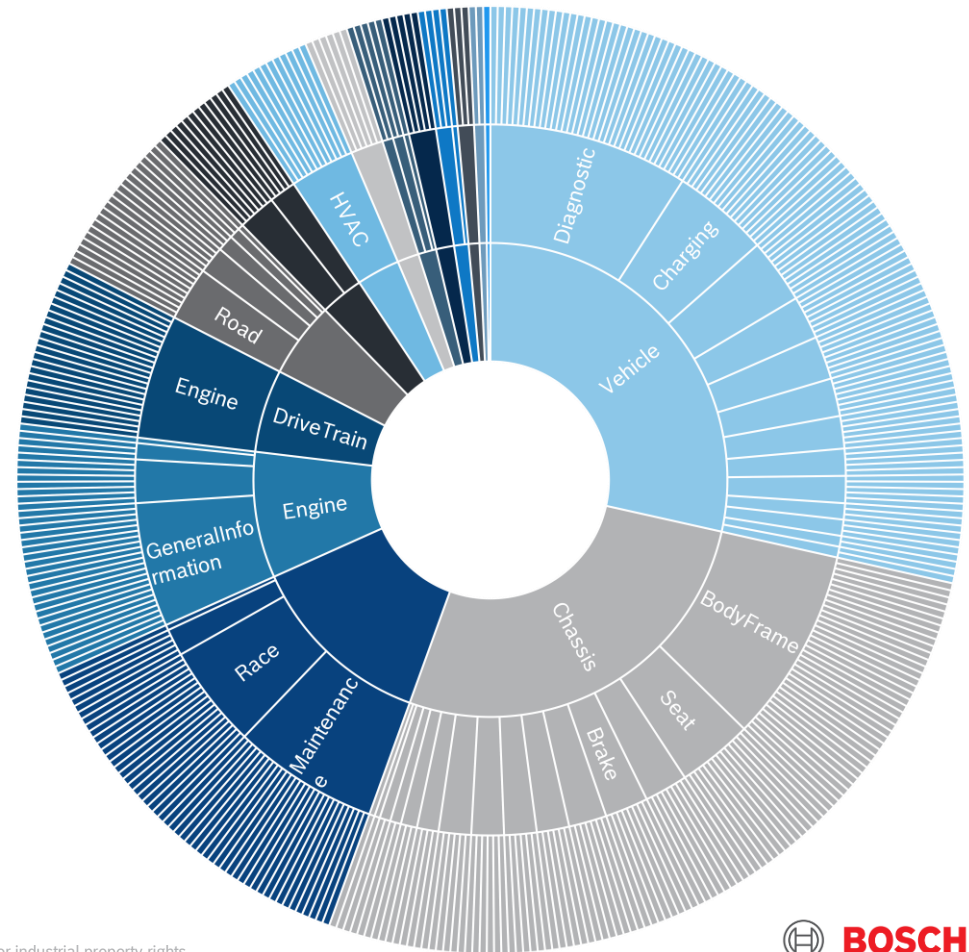
Remaining Challenge: NO Standardized Application Interface over OEMs / Project Borders

# Vehicle Application Plugin Platform

## VAPP Data Model – based on well defined standards



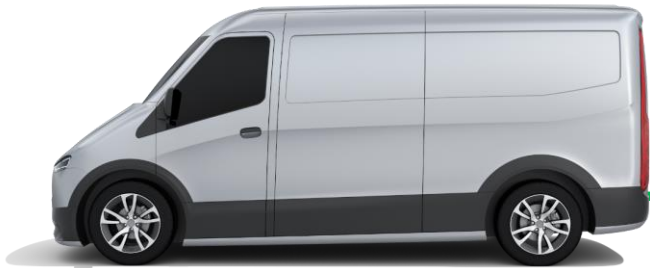
Specific data model = Vehicle + Profile + Extension  
 (e.g. ebike) (e.g. EV)



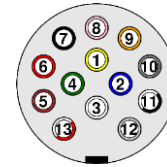


# Vehicle Application Plugin Platform Harmonization Drives Possibilities

A standardized trailer hitch allows a variety of vehicle models to tow a variety of trailers.



ISO11446



By establishing a SW interface standard, VAPP wants to enable a variety of vehicle models to make use of a variety of high-level application software.

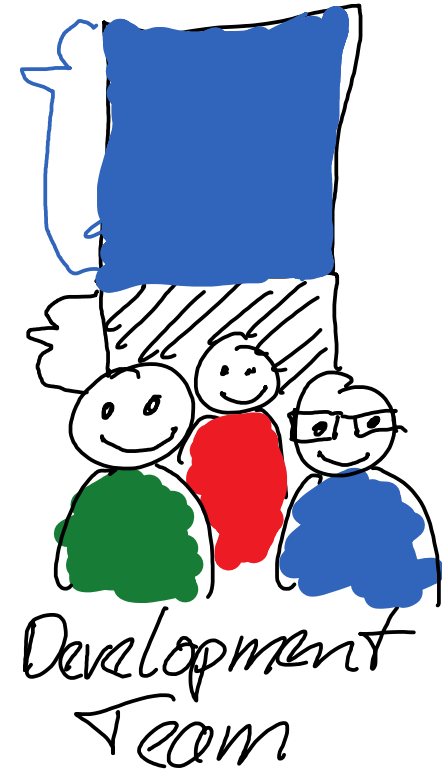


# Vehicle Application Plugin Platform

## Chances in Limiting Interface Variants



Time for cool  
Functionality  
Implementation  
Effort for  
Interface Variants



**Reducing interface variants frees development capabilities for new features!**

# Thank you!



**Dr. Christian Kerstan**

Technical Strategy, Sector Electric & Electronic (M/NET)



Tel. +49 711 811-24406 | Mobile +49 173 7593331



[christian.kerstan@de.bosch.com](mailto:christian.kerstan@de.bosch.com)

