DNA for Autonomous Driving
Porting Robot Architectures to AUTOSAR

Dr.-Ing. Björn Giesler

Apr 14, 2016

Elektrobit

An ACC Developer‘s World in 2002.

Radar

target vehicle candidates,
their velocity / acceleration

Steering Wheel Angle

target vehicle selection

Motor Speed Control

ego vehicle speed control

Head Unit

system activation,
status communication

Items to specify: 4.

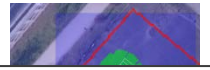
A Driverless Car Developer‘s World in 2016.

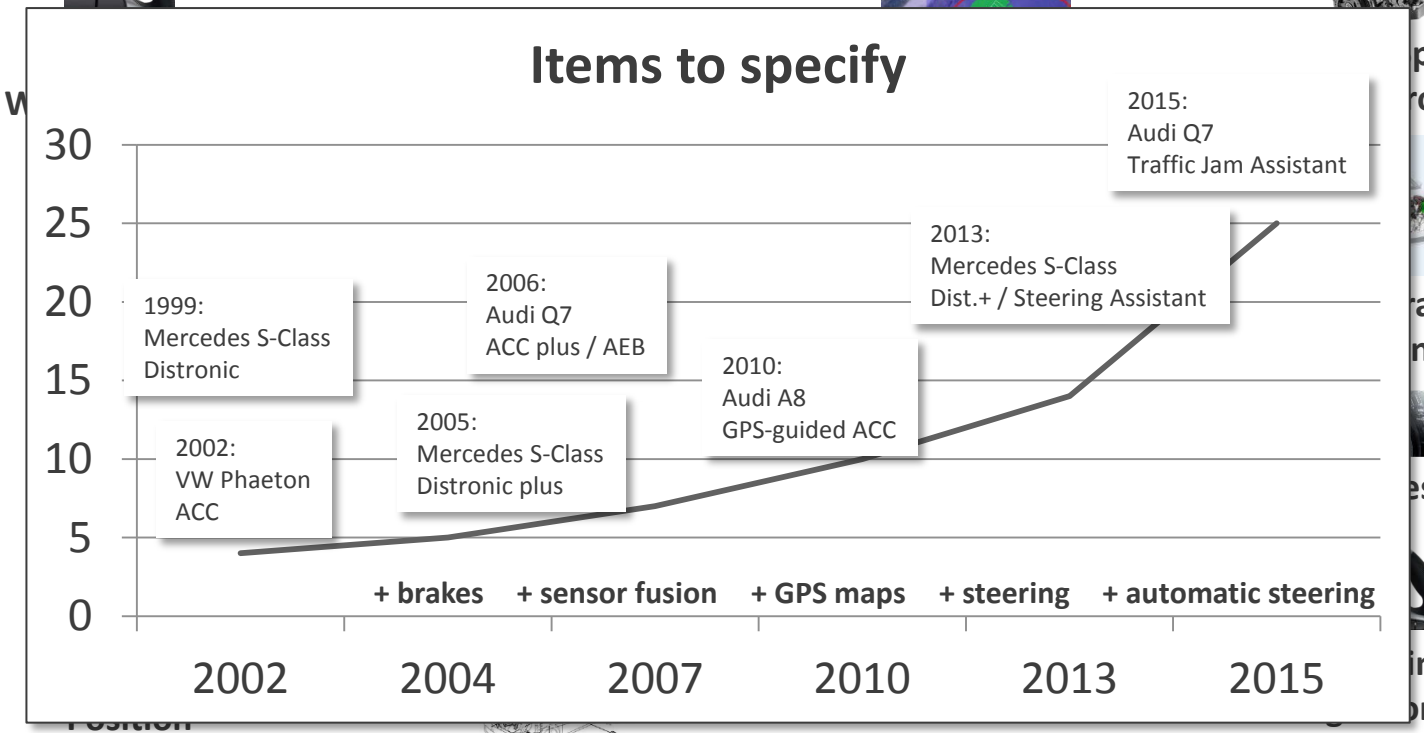
Radar

Camera

LIDAR

Sonar



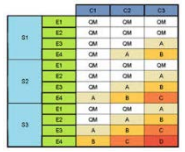


Maps

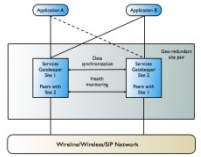
Vehicle
ego motion

Head Unit

Side Tasks



Functional
safety

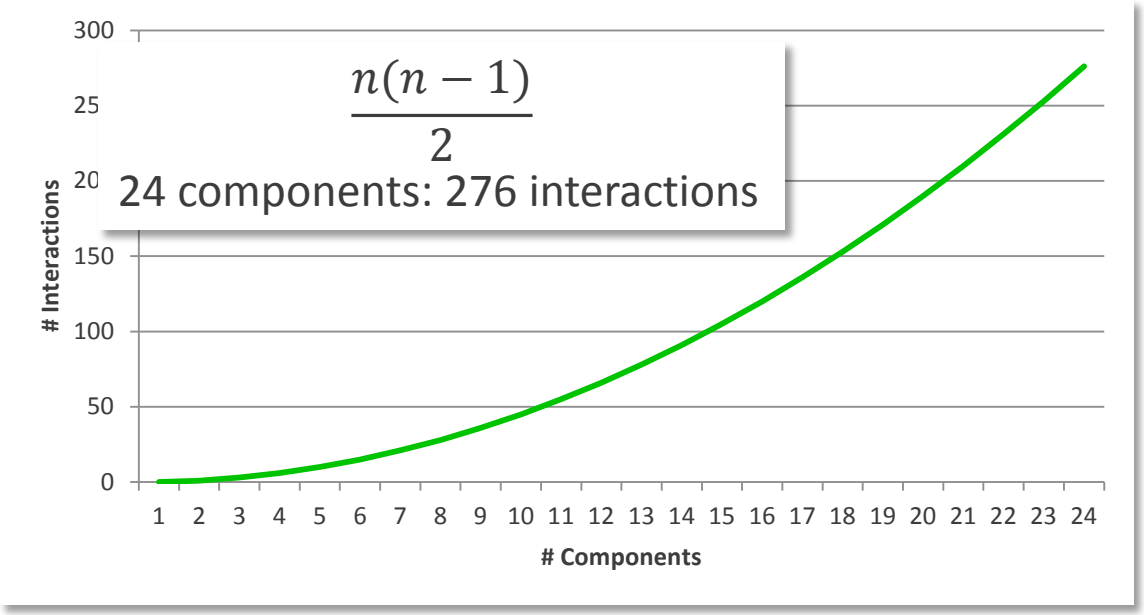


Redundancies

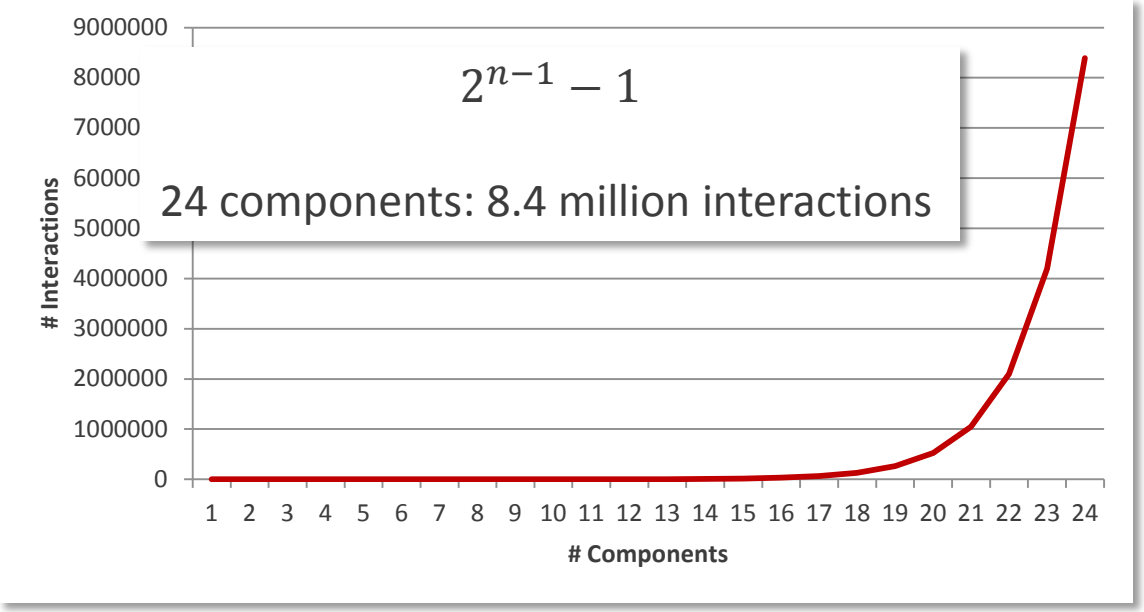
Items to specify: 24.

How Many Interactions Can We Expect For n Components?

Best case:
Every specification gets written or checked only once



Worst case:
All specifications must be re-examined after one is changed



Does A Centralized ECU Help?


Radar


Camera


LIDAR


Sonar


Steering Wheel Sensors

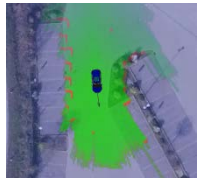

Wheel Speeds


IMU / Gyro

Global Position

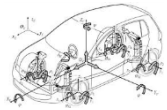

Maps

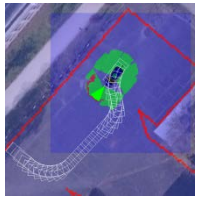

Moving objects


Static obstacles


Lanes

Signs


Vehicle ego motion


Trajectory planning & control

Fail-safe / -degraded mechanisms


Motor Speed Control

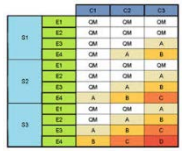

Regenerative Braking

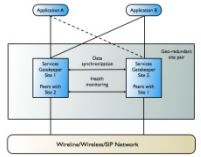

Brake Pressure


Steering Angle Control


Head Unit


Side Tasks


Functional safety


Redundancies

Items to specify: 24.

A Centralized ECU Makes The Problem Worse.


Radar


Camera


LIDAR


Sonar


Central ECU


Steering Wheel Sensors

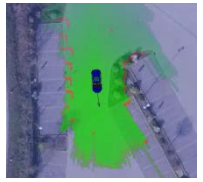

Wheel Speeds


IMU / Gyro

Global Position

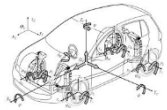

Maps

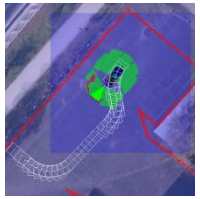

Moving objects


Static obstacles


Lanes

Signs


Vehicle ego motion


Trajectory planning & control

Fail-safe / -degraded mechanisms


Motor Speed Control

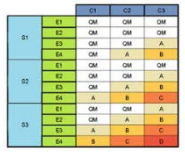

Regenerative Braking

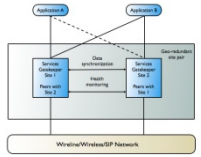

Brake Pressure


Steering Angle Control


Head Unit

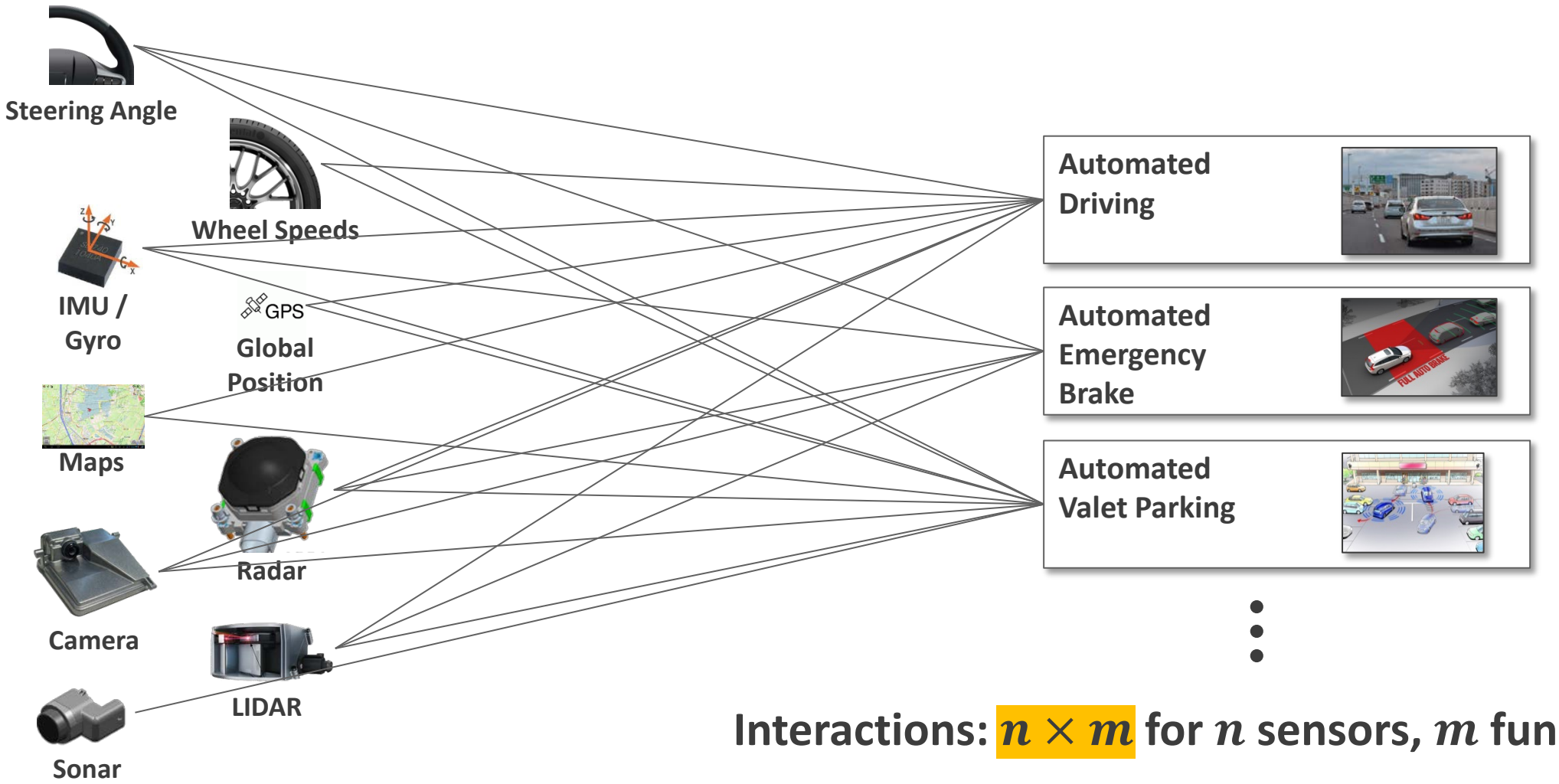

Side Tasks


Functional safety


Redundancies

Items to specify: 25.

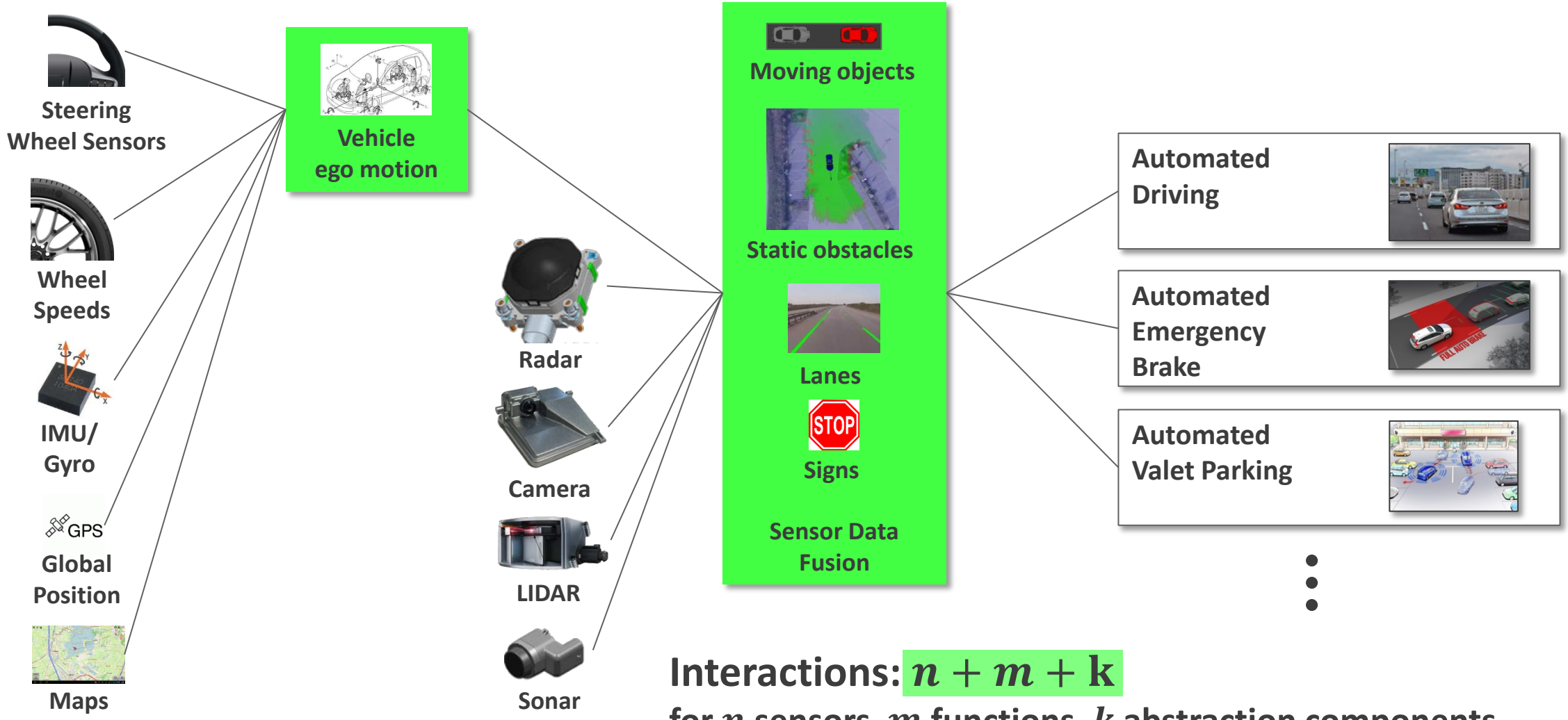
So What Does Help?



Interactions: $n \times m$ for n sensors, m functions

Elektrobit

So What Does Help?



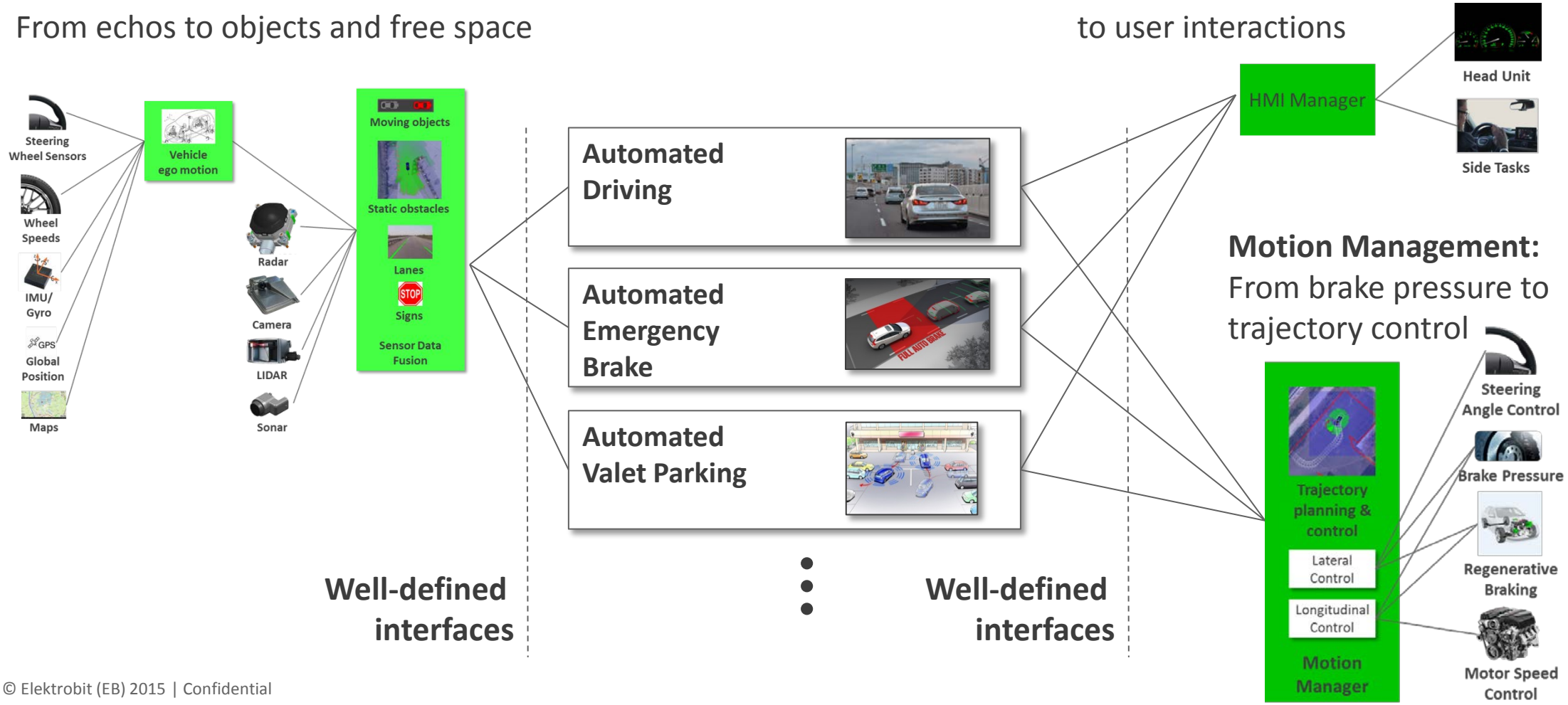
Interactions: $n + m + k$

for n sensors, m functions, k abstraction components

From Exponentiality to Linearity

Sensor Data Fusion:
From echos to objects and free space

HMI Management:
From buttons and LEDs
to user interactions



Functional Architectures

Can we make complicated behavior even simpler?

Automated Driving

Drive in your own lane!

Change lanes if you must!

Don‘t crash!
(for heaven‘s sake!)

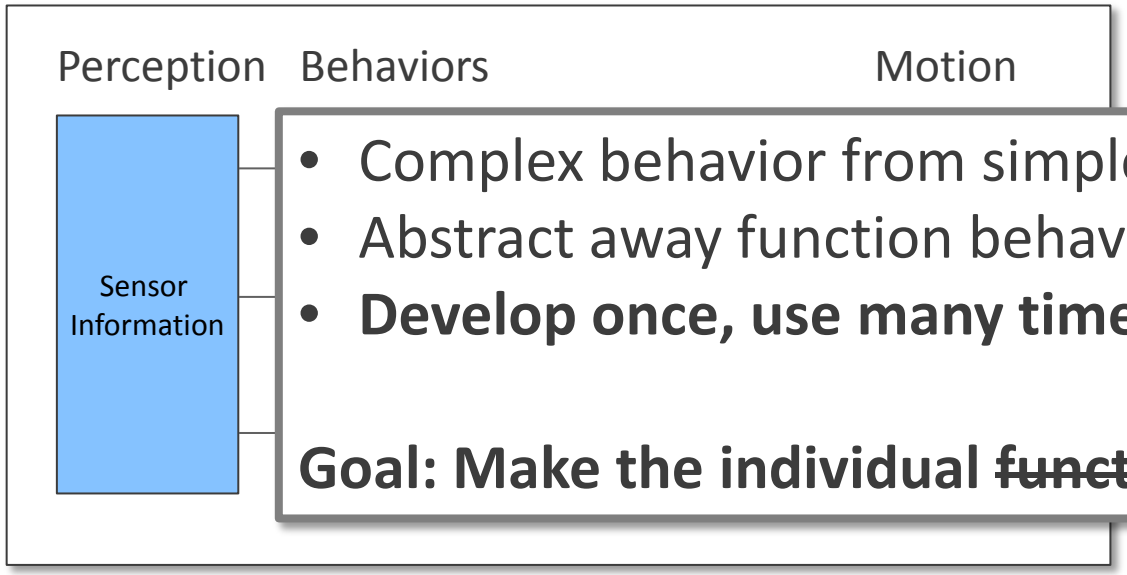
Come to a safe stop if necessary!

Continue on, or switch lanes?

Swerve, get back in lane, or brake?

Is everything still OK?

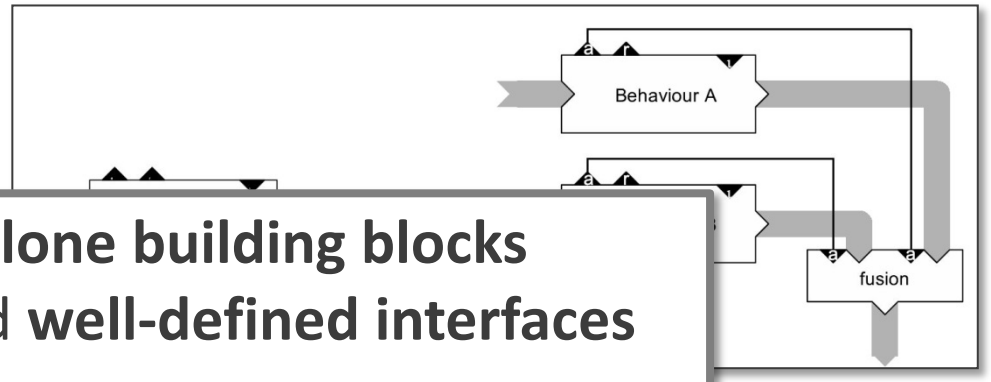
Robot Architectures

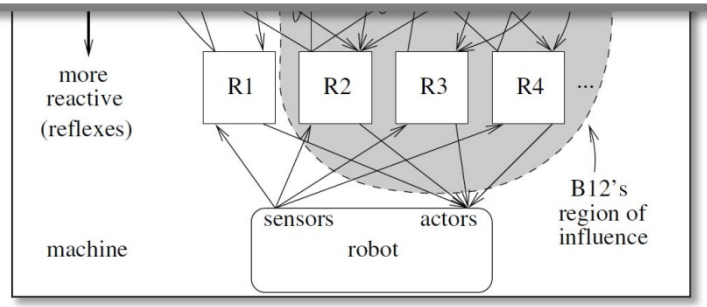


- Complex behavior from simple, **stand-alone building blocks**
- Abstract away function behavior behind **well-defined interfaces**
- **Develop once, use many times**

Goal: Make the individual ~~function~~ behavior as simple as possible!

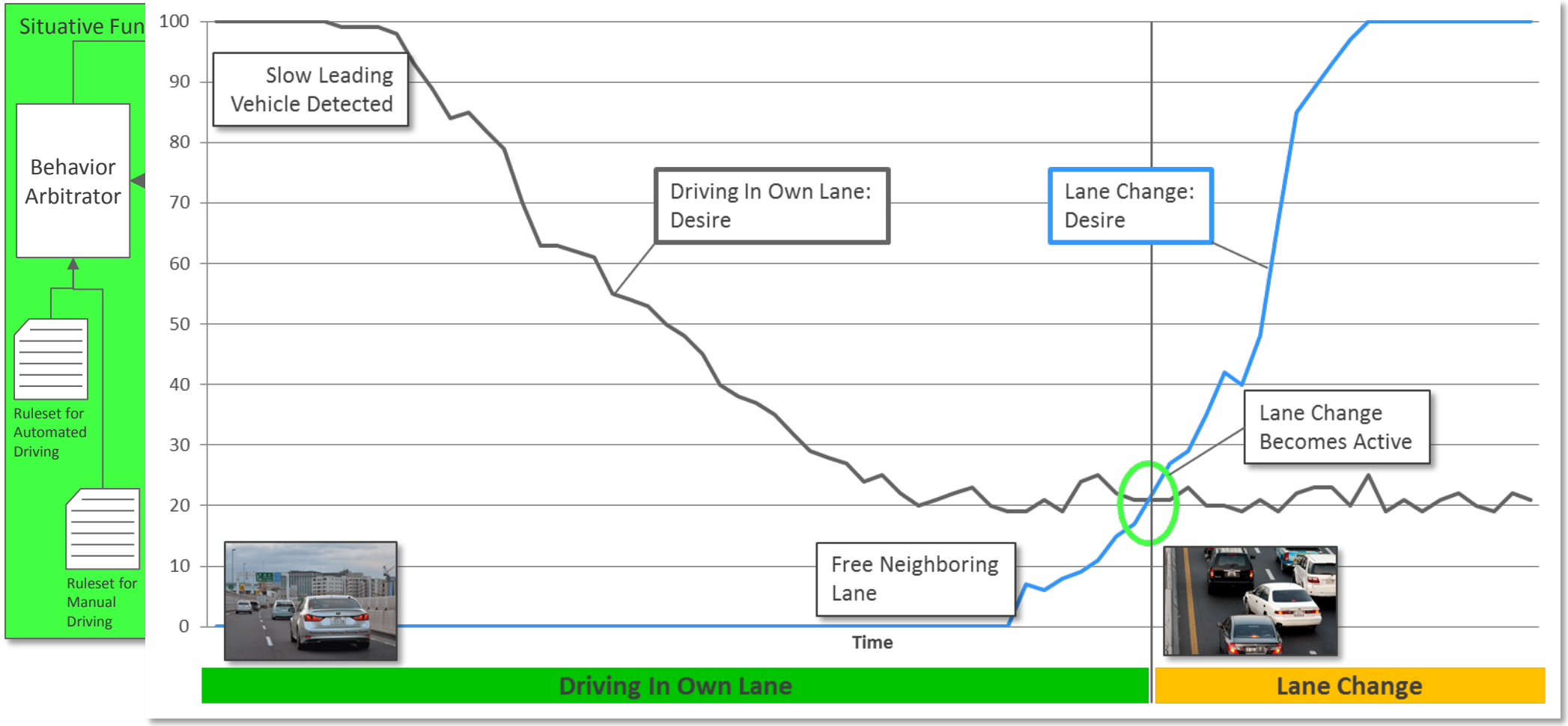
Subsumption Architecture
R. Brooks, MIT, 1986





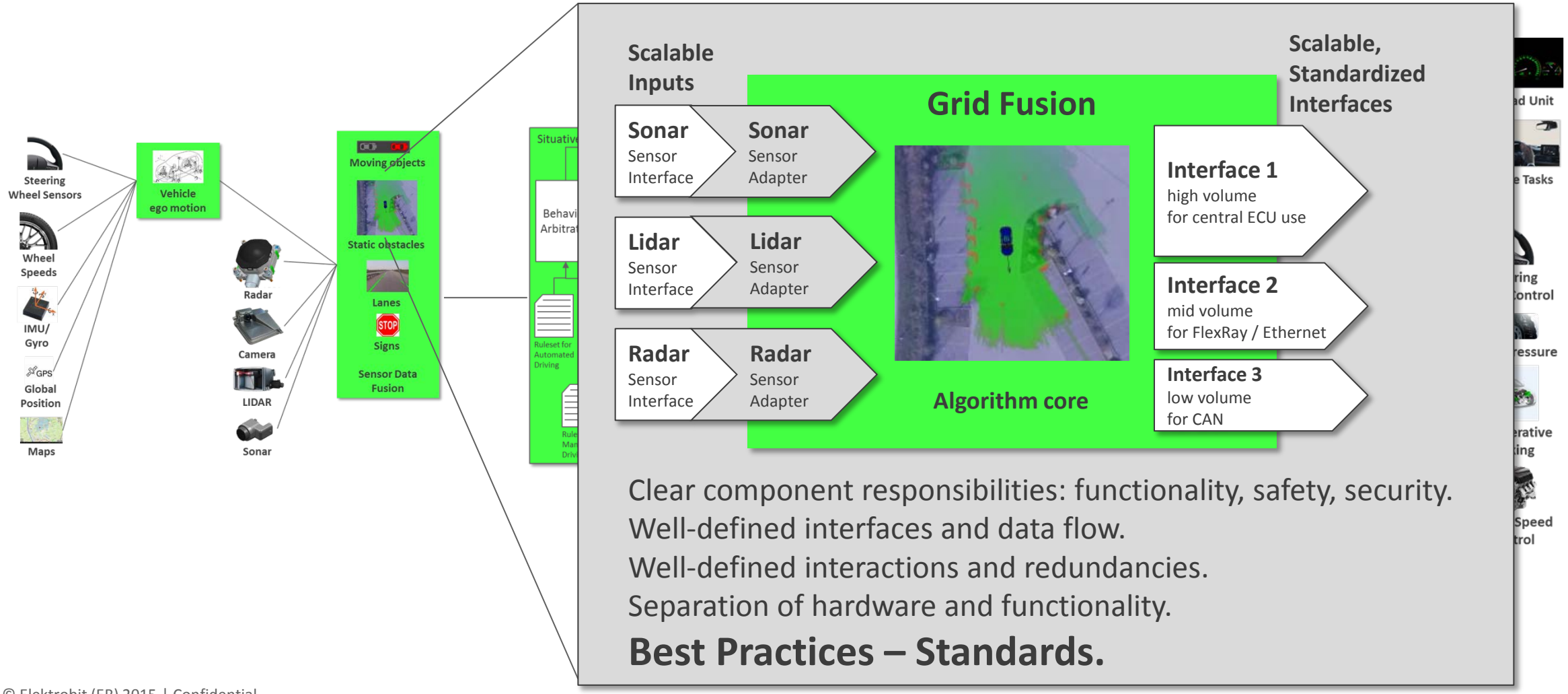
Behavioral Networks
J. Albiez / R. Dillmann, KIT, 2006

Robot Architectures for Highly Automated Driving



Elektrobit

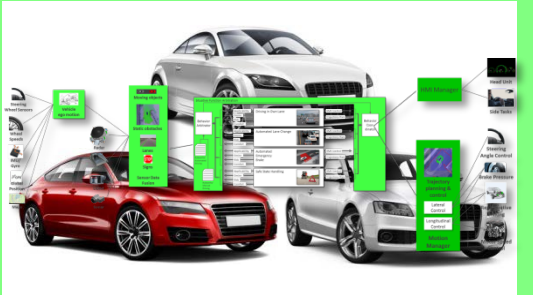
Architecture Is Key To Managing Complexity.



Best Practice Leverage: Project, Product Family, Industry

Project-Level Architecture Best Practices:
Reduce complexity

€



Product Family Architecture Best Practices:
Re-use specified, developed,
industrialized and tested components

€€

Industry-Wide Architecture Best Practices
Peer review of functionalities, safety and security mechanisms
Lower entry into HAD development
No vendor lock-in

€€€

Elektrobit

Conclusion

The problem is not difficulty but complexity.

Good functional architectures help solve it.

We should talk about software, not hardware architectures.

We should move towards best practices across the whole industry.

Less work, less cost, more competitiveness and better functions.

Thank you.

Elektrobit

www.elektrobit.com
bjoern.giesler@elektrobit.com