

Software-defined Vehicles

Some observations on how to make the software factory work.

Oliver Bahns, Head of Connected Mobility, T-Systems

T Systems

Let's power
higher performance

Connected Mobility @ T-Systems

We build and operate connected car platforms



Multi-Cloud provider



> 25 **mio**
vehicles

under management

worldwide, 24/7

with > 30 MNOs
and content providers

Global platform

200 supported countries worldwide
> 50.000 requests per second per
microservice

> 10 years

Expertise in Connected Mobility

DevOps & Cloud migration

experience for connected backends:

seamless cloud migration

< 5 months

Cloud-agnostic

approach based on Building Blocks

T-Systems Hypercube: Production-ready software components for software-defined vehicle.



Device & Campaign Management

Management of software & configuration, inventory, fleets and user & access



Digital Twin

Representation of an individual vehicle or fleet to facilitate data-driven services



Vehicle Operations Center

One-stop shop for fleet analysis and end-to-end lifecycle management of connected vehicles



Task Management

Low-code solution to create, adapt and implement process workflows



Packaging Service

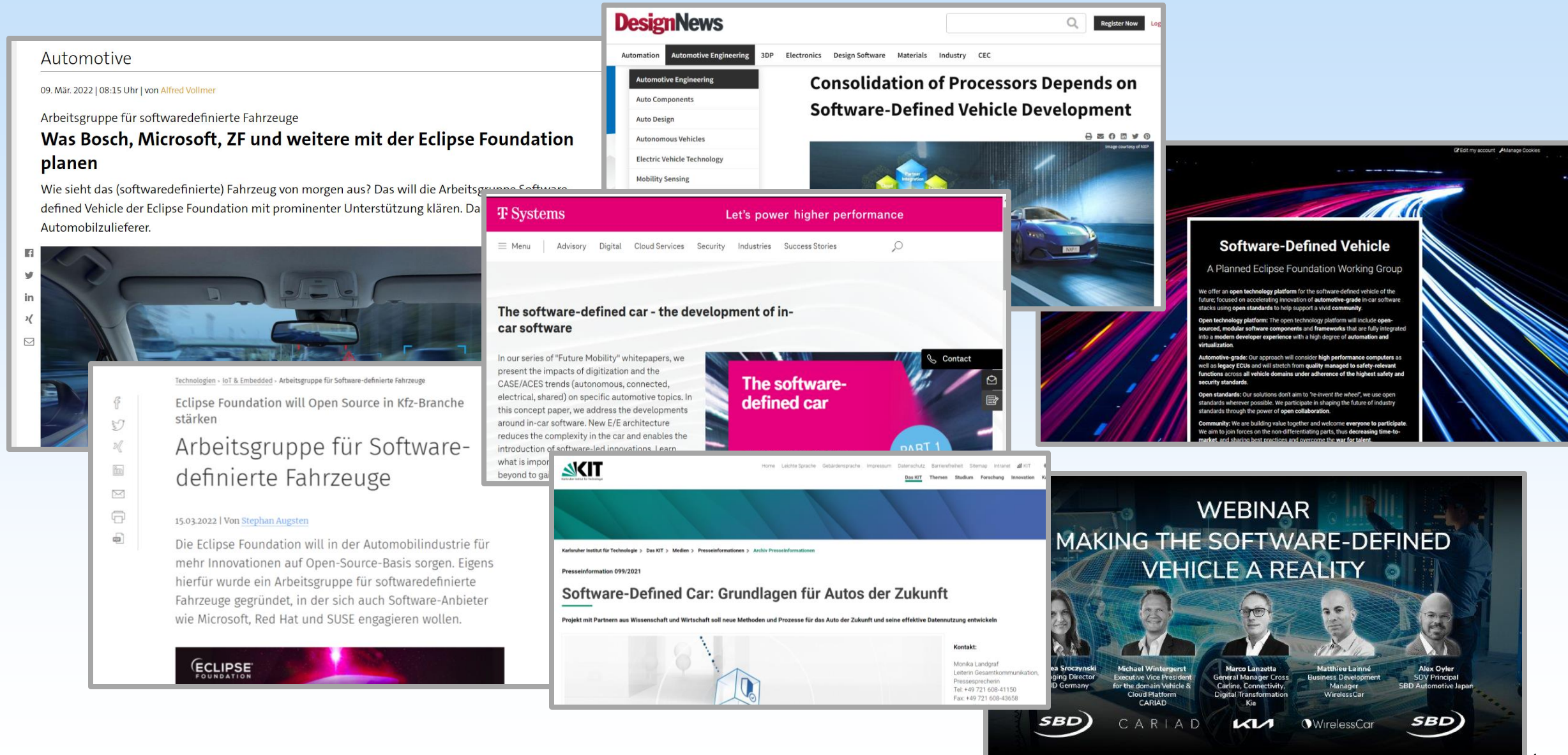
Automated, quality-assured and secured assembly of vehicle software packages



Developer Console

Active management of the complete software lifecycle (development, testing and release)

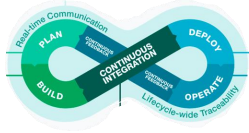
SW-defined Vehicle – the new acronym for CASE?



Car Manufacturers transform into software-driven organizations by implementing an end2end software perspective.

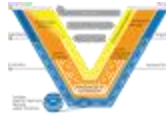
Connected Services & Infotainment

Domain-specific process & tool chain



Body & Chassis

Domain-specific process & tool chain



ADAS

Domain-specific process & tool chain



Automated Driving

Domain-specific process & tool chain



Four fundamental consequences of the strategic transformation:

Consequent de-coupling of HW and SW

Consistent lifecycle management for all vehicle software (single source)

CI/CD Software Pipeline and Data Loop between vehicle, edge, cloud

Service-oriented and DevOps driven organization

While we see different strategies in our industry, there are 4 common ingredients:

1

Come to a Product focus

2

Manage the output, not the people

3

Enable the Ops in DevOps

4

Move to “Build, Buy and Compose”

1 Product Focus

Given the different life cycles, **everything is a product** with a defined and committed feature set!

From feature to version, from version to roadmap enables planning beyond next sprint or program increment.



Managing the output, creates ownership beyond coding.

DevOps team members become the Architects of Outcome

together with product management, brand and market teams (... up to removing the classic 1st, 2nd, 3rd level of support).

1 Product Focus

Sounds easy, BUT ...
needs the right strategy
per product!



- **Partnering** is just one approach – will all partners have the same product mindset and commitment?

→ Probably no ...

- **Fully managed platforms** work well, if well defined and integrated across products.

→ Not flexible enough

- **Do it yourself** requires time, capacities and skills.

→ Rarely available ...

2 The „Army of People“ problem

Shortage of capacities and skills demands for alternatives by standardization and automation:

- Standardization and a commodity software approach
- Converged platforms to integrate platforms, tools and services

▪ **Get the people, keep the people, create the outcome!**

▪ **Productivity needs structure:**
too many bright people can not form high performing teams.

▪ **From vision to execution:**
Output of 4000 people can be the output of 1000, if poorly managed.

... and don't forget to keep your teams happy –
let them do what they can do best,
remove the rest ... or automate it!

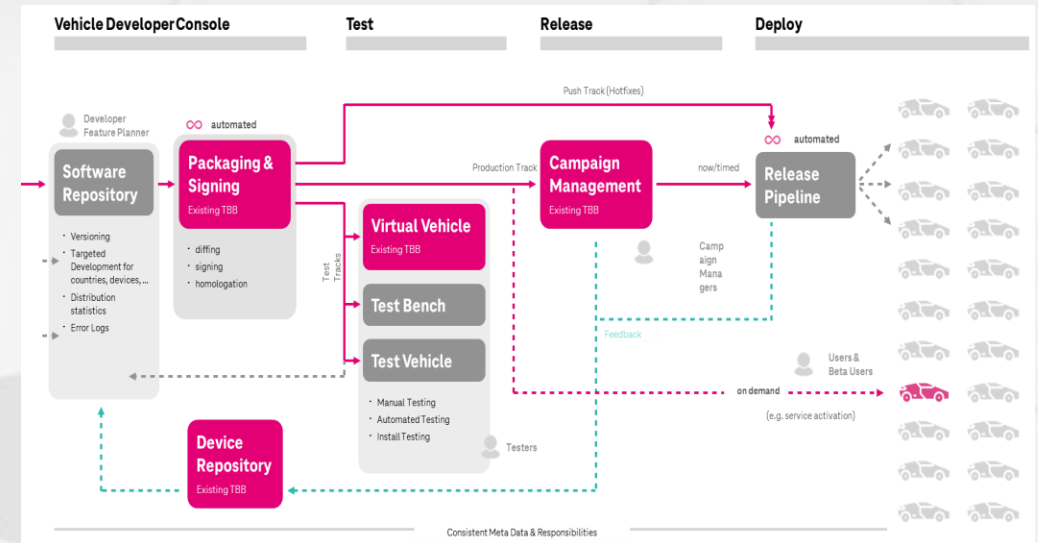
3 The missing Puzzle Pieces of DevOps

The Software Continuum starts with DevOps, **but ...**

- Everyone claims to have it – still with a **massive dev-only scope**
- Aside of the SW-defined vehicles and related platforms, **significant legacy and current platforms exist and will co-exist for a long time.**
- “Automate everything”, but **“Operate anywhere”** needs to be added -> integral part of the Software Continuum.

DevOps along the Software continuum will be a mesh of distributed responsibilities with a product view –

RUN the product for millions of vehicles • • • across platforms and technologies ...



Vehicle Developer Console: Integration across the SW Continuum

... and don't forget data/process governance, regulatory/local requirements and privacy & GDPR.

4 Build, Buy and Compose

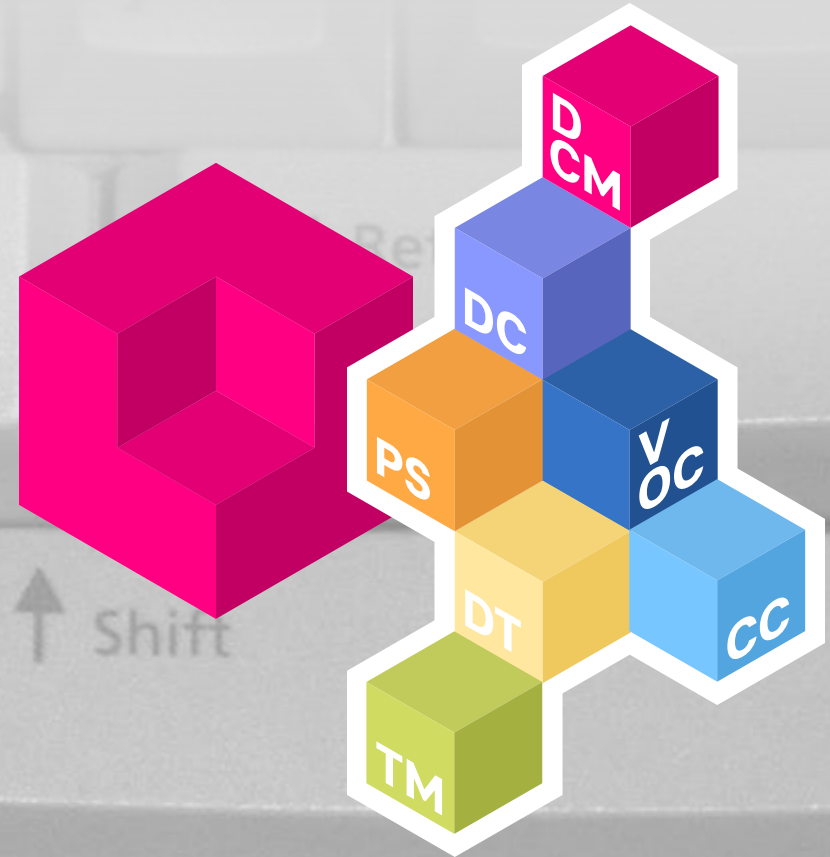
Make or Buy is out.

Let's apply the principles of “**Build, buy and compose**” to the Software Continuum:

- **Buy** commodity
- **Built** the differentiating parts
- **Compose into a solution**

Composable thinking, composable business architectures, composable technologies

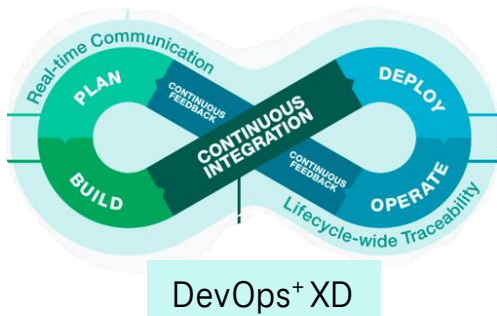
are the building blocks based on known concepts like modularity, discovery, orchestration and resilience through autonomy.



Hypercube – central backbone for cross-domain automotive software lifecycle management

Connected Services & Infotainment

Domain-specific process & tool chain



Body & Chassis

Domain-specific process & tool chain



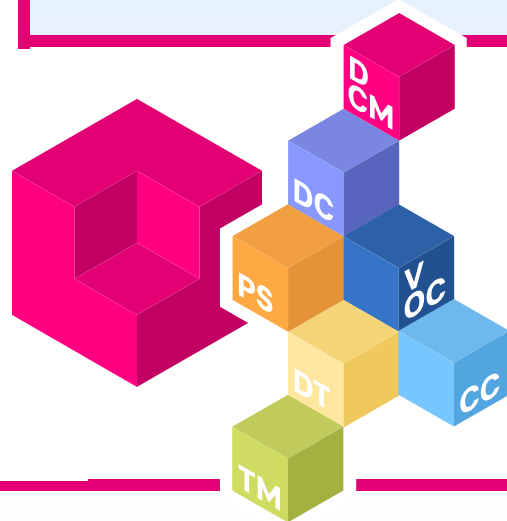
ADAS

Domain-specific process & tool chain



Automated Driving

Domain-specific process & tool chain



Hypercube enables:

- DevOps for vehicle software
- Data-driven software development across multiple vehicle domains
- Operation of software defined vehicle fleets
- Data-enabled insights/services

In summary ...

The ingredients are clear,
the success comes with the detailed recipe,
the preparation and the actual doing.

- 1 The definition of a product is multi-dimensional.
- 2 Resources remain limited – efficiency and output will be key.
- 3 Worldwide operations of software is needed.
- 4 No-one can do everything – use standards and components wherever possible. Composing the elements creates the IP.



Source: Infineon

Thank you!