

SDV: Overview

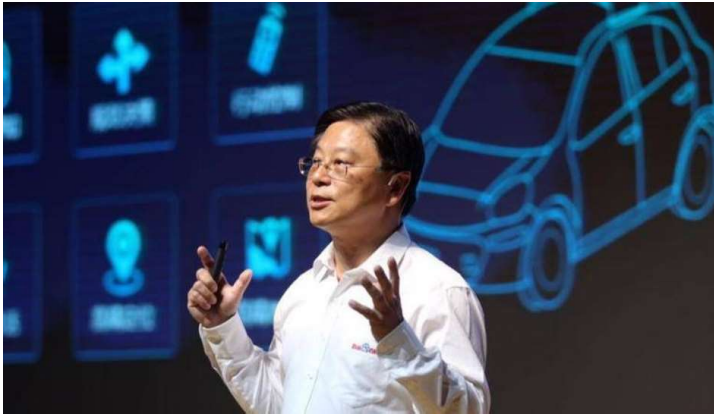
1. Contents

- 1. Contents
- 2. Definition
- 3. Driving forces
 - 3.1. ACES (Automated, Connected, Electrified, Shared)
 - 3.2. Centralized Domain Control
 - 3.3. Software Complexity
 - 3.4. The Success of Tesla
- 4. Technologies
- 5. Changes
- 6. Benefits
- 7. Conclusion
- 8. References

2. Definition

SDV (Software-defined vehicles) is a term that describes a vehicle whose features and functions are primarily enabled through software, a result of the ongoing transformation of the automobile from a product that is mainly hardware-based to a software-centric electronic device on wheels.

"Vehicles that dominate the future will be software defined vehicles (SDVs) based on artificial intelligence, and the ultimate state would be driverless vehicles." said Wang Jing, senior VP of Baidu, 2016.



"The car will become the most complex internet device we have known so far, the car will become a software product." said Herbert Diess , Chairman of the Board of Management of Volkswagen Group, 2019.



The idea of tomorrow's car as a computer on wheels is ripening apace.

3. Driving forces

3.1. ACES (Automated, Connected, Electrified, Shared)

Four powerful trends — electrification, automation, connectivity, and sharing — are reshaping customer expectations and driving manufacturers to increasingly turn to software to address them.

Software enables critical automotive innovations.

ware Defined Vehicles)

Software innovation examples

Connectivity

- Integration of 3rd-party services
- Updates over the air to deploy new features faster
- Operation of future cars partly in the cloud

Autonomous driving

- Rise of built-in sensors and actuators
- Higher demand for computing power and communication
- Unlimited need for reliability

Electrification

- Introduction of new electronics
- Reduction of energy consumption through advanced software algorithms

Diverse mobility

- Shared-mobility services and robo-taxis via app
- Customized driver experience

Innovation through software

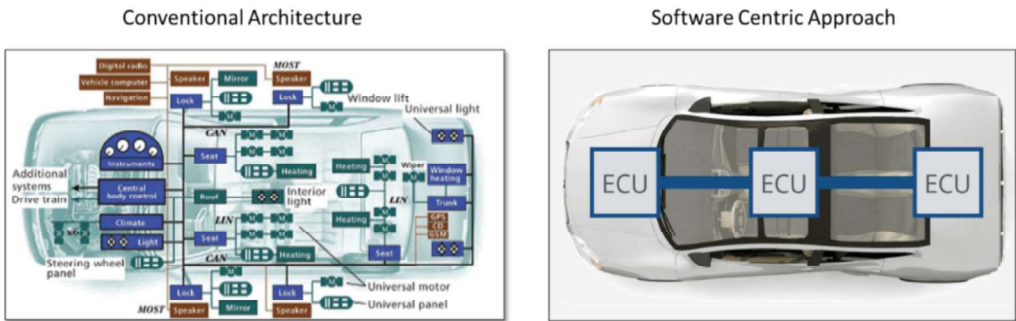
Source: Automotive Electronics Initiative; HAWK; IEEE, "This car runs on code"; McKinsey analysis

McKinsey&Company

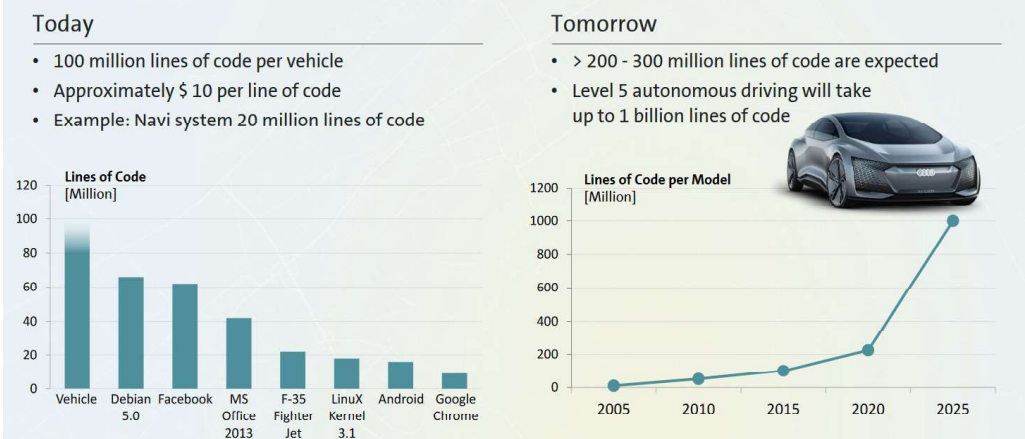
As the car continues its transition from a hardware-driven machine to a software-driven electronics device, the auto industry's competitive rules are being rewritten.

3.2. Centralized Domain Control

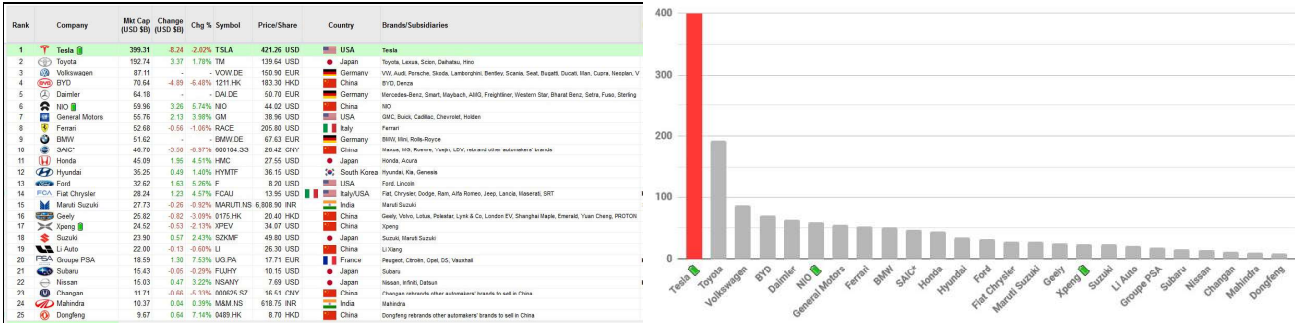
While the distributed systems of vehicles have been practical and functional up until now, newer architectures, more data, and a movement toward centralized processing is challenging the limits of traditional automotive design. One such trend is the consolidation of domains into a centralized computer architecture. Instead of having several discrete ECUs, they are all housed in a centralized domain control. This model allows for greater software functionality, more code reuse, and easier remote management.



3.3. Software Complexity




3.4. The Success of Tesla



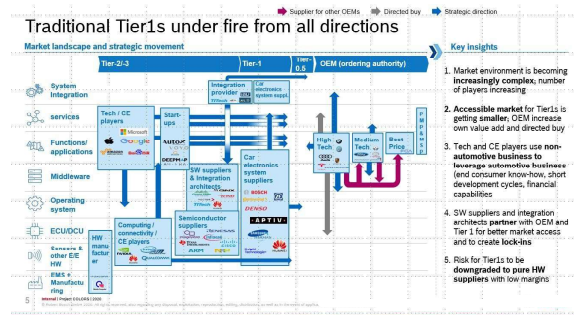
4. Technologies

The car becomes the most complex internet device.

Security	Cloud	Safety
Operating Systems Audio AI IoT Computing Autonomous Self-driving AutoSAR Hypervisor Diagnosis Graphics		HMI Optical EE Mechanical Web P2P Navigation AVB FM/AM DA Most
V2X, 5G, CCU	SoC, MCU	CAN, Aeth, A2B

5. Changes

- A key piece in the evolution of the software-defined vehicle will be the separation of software and hardware development.
- Instead of a development cycle focused on "model years," agile methodologies will drive continuous software development, and OEMs will be able to deploy software to the vehicle even after it has left the factory.
- Computing demands will increase, as vehicles process data from various sensors and interact with a broad ecosystem that includes other vehicles on the road. Vehicle manufacturers will have to develop data-analytics systems capable of handling this vast data flow and processing it in near real time.
- The industry will rethink its approach to vehicle software and electrical and electronic architecture, moving to a more modular service-oriented architecture (SOA) model, which makes it easier for software components to be reused in a building-block format.
- Security strategies will become even more critical to avoid, detect and defend against cyber attacks, and those strategies will have to evolve to protect the whole system rather than simply protecting individual components or boxes within it.
- OEMs start to ramp up software competency and tend to develop their own software stacks.
- Traditional Tier1s under fire from all directions:



6. Benefits

- Customers will be able to receive over-the-air (OTA) updates that cover security patches, infotainment improvements, plus monitoring and tuning of core functional capabilities of the vehicle, such as power-train and vehicle dynamics.
- OEM can lock/unlock new safety, comfort and convenience features on the fly
- Giving vehicle manufacturers insight into every aspect of a vehicle, its performance and its place in the connected ecosystem
- OEM has opportunity to improve life-cycle management and develop revenue-generating features they can offer to customers
- Create new business models and revenue streams such as theft-prevention services, emergency assist alerts, travel guides, on-demand services/features.

In short the software-defined vehicle will create opportunities for both consumers and OEMs, many of which haven't even been conceived yet.

7. Conclusion

Up to now we can see there are three typical modes to respond to this trend.

Mode	Type	Follower	Aim	Label	Characteristics	Status
Apple-mode	Tesla	NIO, Xpeng, VW, SAIC	Provide excellent product for end customers	Product	<ul style="list-style-type: none"> • Close product • Iterating hardware, software, and services among products • Technological advance sensitive 	Partial successful
Android-mode	Huawei?	Baidu, Banma	Provide powerful and flexible platform for partners	Platform	<ul style="list-style-type: none"> • Close platform and open ecosystem • Iterating platform among internal release • Ecosystem and user number sensitive 	Trying
Design-house-mode	Bosch?	Aptiv, Joynext, and other traditional Tire1/2	Provide competitive engineering solutions for OEMs	Project	<ul style="list-style-type: none"> • Open solution • Iterating software components and engineering competency among projects • Cost and competency sensitive 	Ongoing

8. References

1. <https://www.apativ.com/insights/article/what-is-a-software-defined-vehicle>
2. <https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/rethinking-car-software-and-electronics-architecture#>
3. <https://www.autovision-news.com/hmi/vehicle-control/changes-vehicle-electrical-architecture/>
4. <https://auto.sina.com.cn/jishu/2020-09-01/detail-iivhvpwy4212861.shtml>
5. https://www.volkswagenag.com/presence/investorrelation/publications/presentations/2019/03_march/1_CMD_Diess.pdf
6. <https://alltechasia.com/baidu-to-launch-driverless-cars-in-scenic-wuzhen/>
7. <https://www.linkedin.com/pulse/car-become-software-product-herbert-diess/>
8. <https://docs.google.com/spreadsheets/u/1/d/1HfIVng6sYlb6Gs4pOKiDGtqU5YJ2-hgdM4pRNaT62gs/htmlview>
9. <http://stock.tianyancha.com/ResearchReport/eastmoney/5348e9b42159c8a0c7c7edcc8b5cdab7.pdf>
10. <https://www.vzkoo.com/doc/14690.html?a=1&keyword=%E8%BD%AF%E4%BB%B6%E5%AE%9A%E4%B9%89%E6%B1%BD%E8%BD%A6>

No labels

