Communication Management in Automotive Service-Oriented Architectures

Trista Lin, David Fernandez Blanco, Juleixis Guariguata

IEEE SA Ethernet & IP @ Automotive Technology Day 2021

E/E ARCHITECTURE OVERVIEW

SOA COMMUNICATION PROTOCOLS

TAKEAWAYS & PERSPECTIVE

STELLANTIS

E/E ARCHITECTURE OVERVIEW

SOA COMMUNICATION PROTOCOLS

TAKEAWAYS & PERSPECTIVE

STELLANTIS

Cloud

Apps

OS

Sensors

Actuators

HW

Cloud

Apps

Car.OS
(Multi OS+MW)

Sensors

Actuators

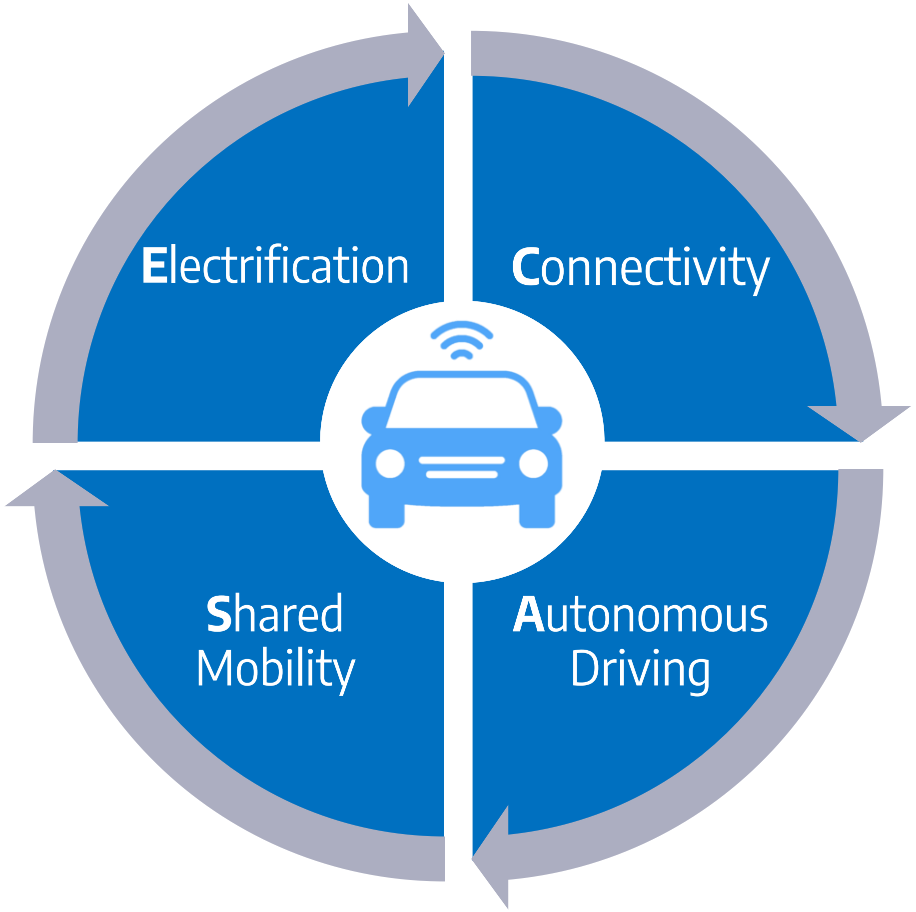
HW

Hardware-defined
Off the lot = Depreciation

Software-defined
Update = Appreciation

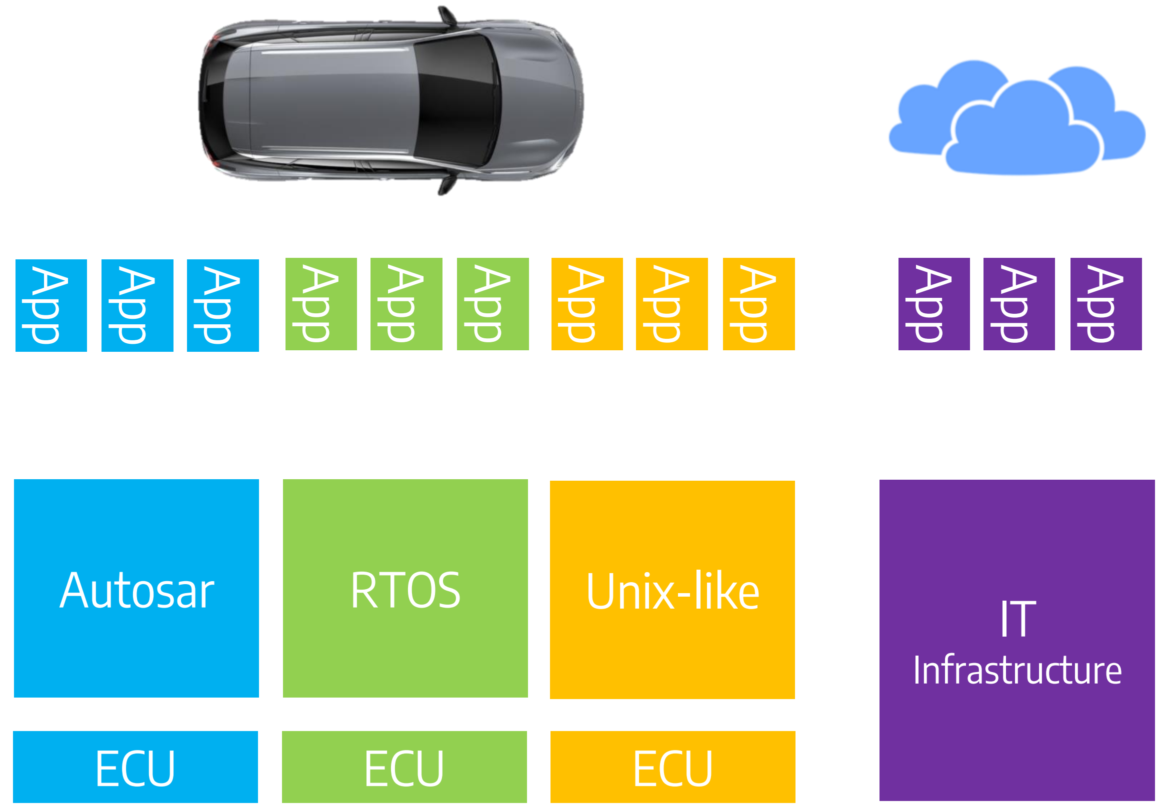
- SDK providing standard APIs
- Software Services
- Infrastructure Virtualization
- Domain → Cross-Domain → Zonal architecture

Service **O**riented **A**rchitecture



- Continuous improvement
  - Connectivity & OTA
- Self or cooperative driving
  - AD/ADAS & V2X
- Cross-domain needs
  - Domain overlapping between ADAS, HMI, energy, connectivity, etc.
- In-car Marketplace
  - Apps and features on demand
- New vehicle ownership
  - Carsharing, user profiling, etc.
- Scalable car platform
  - Seamless integration and software modularity
- Software complexity
  - Decoupling, integration, versioning, variant management, etc.
- Time to market
  - Reduce Complexity and facilitate integration

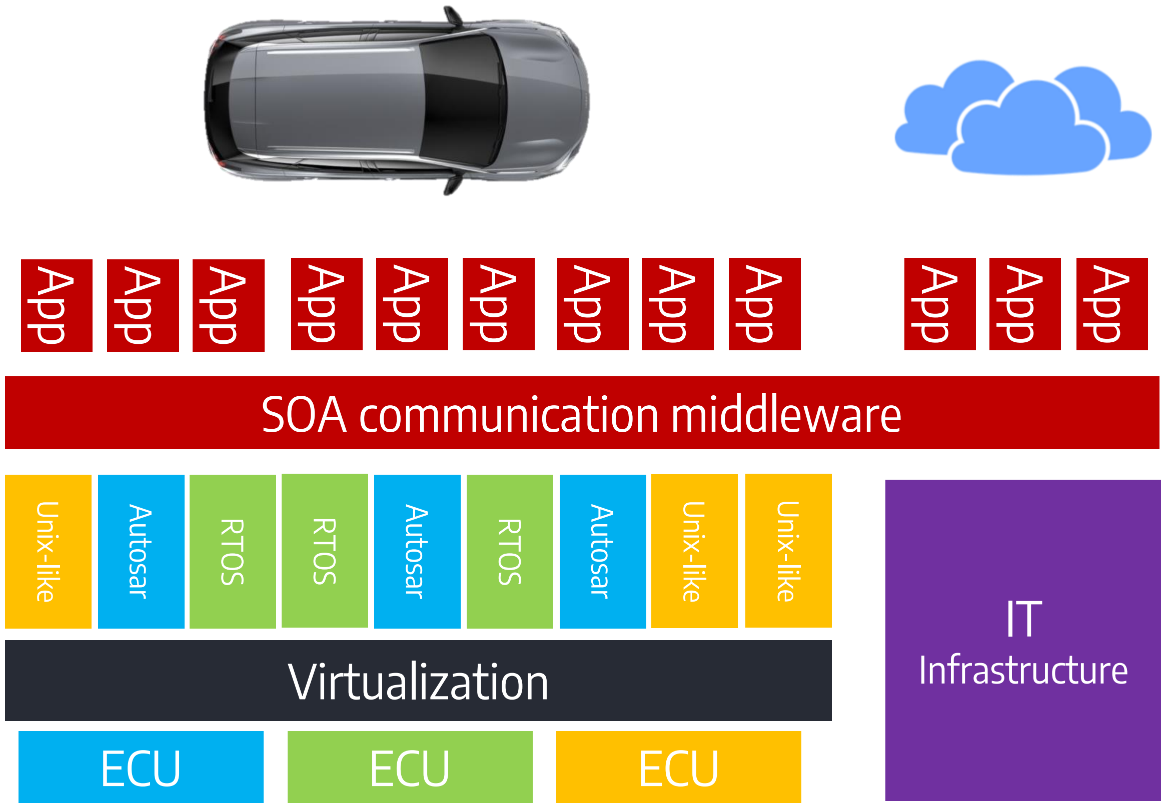
TOWARDS SOA – TARGET



HW abstraction
Resource optimization
Reconfigurability
Simplify:
- Resource management interfaces
- Service allocation

Virtualized E/E Architecture
- System (HW & OS)
- Network

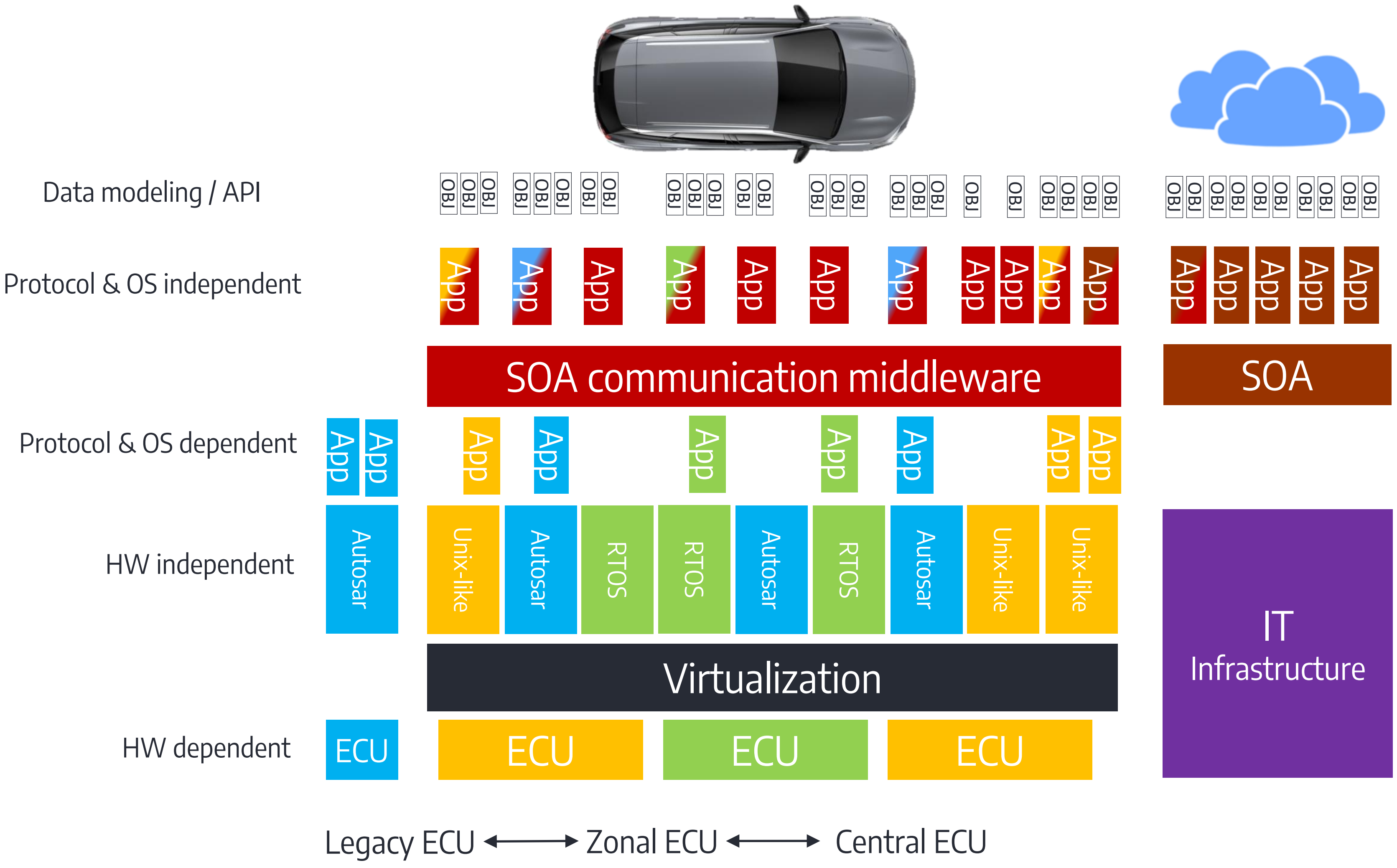
E/E Architecture



From signal to service
Service discovery
HW/SW decoupling

Data-centric
Modularity
Reusability
Simplify:
- New use case adding
- Data interfaces

SOA requires a service-oriented infrastructure



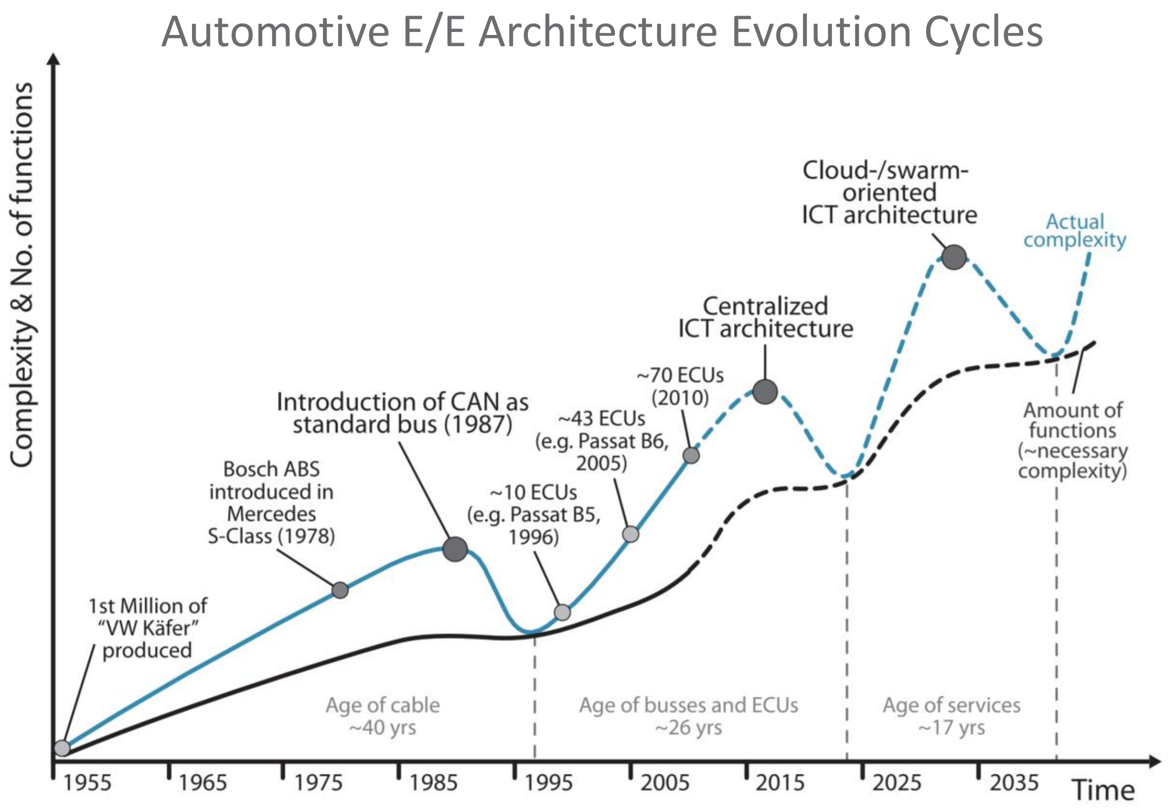
Key points:

- Heterogeneity
- Coexistence
- Boundary
- Standardization

E/E ARCHITECTURE OVERVIEW

SOA COMMUNICATION PROTOCOLS

TAKEAWAYS & PERSPECTIVE



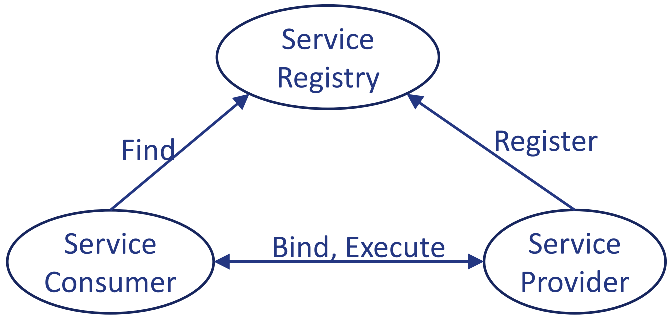
(Source: C. Buckl *et al.*, "The software car: Building ICT architectures for future electric vehicles," *IEEE International Electric Vehicle Conference*, 2012)

Ethernet & IP enable the deployment of centralized architecture

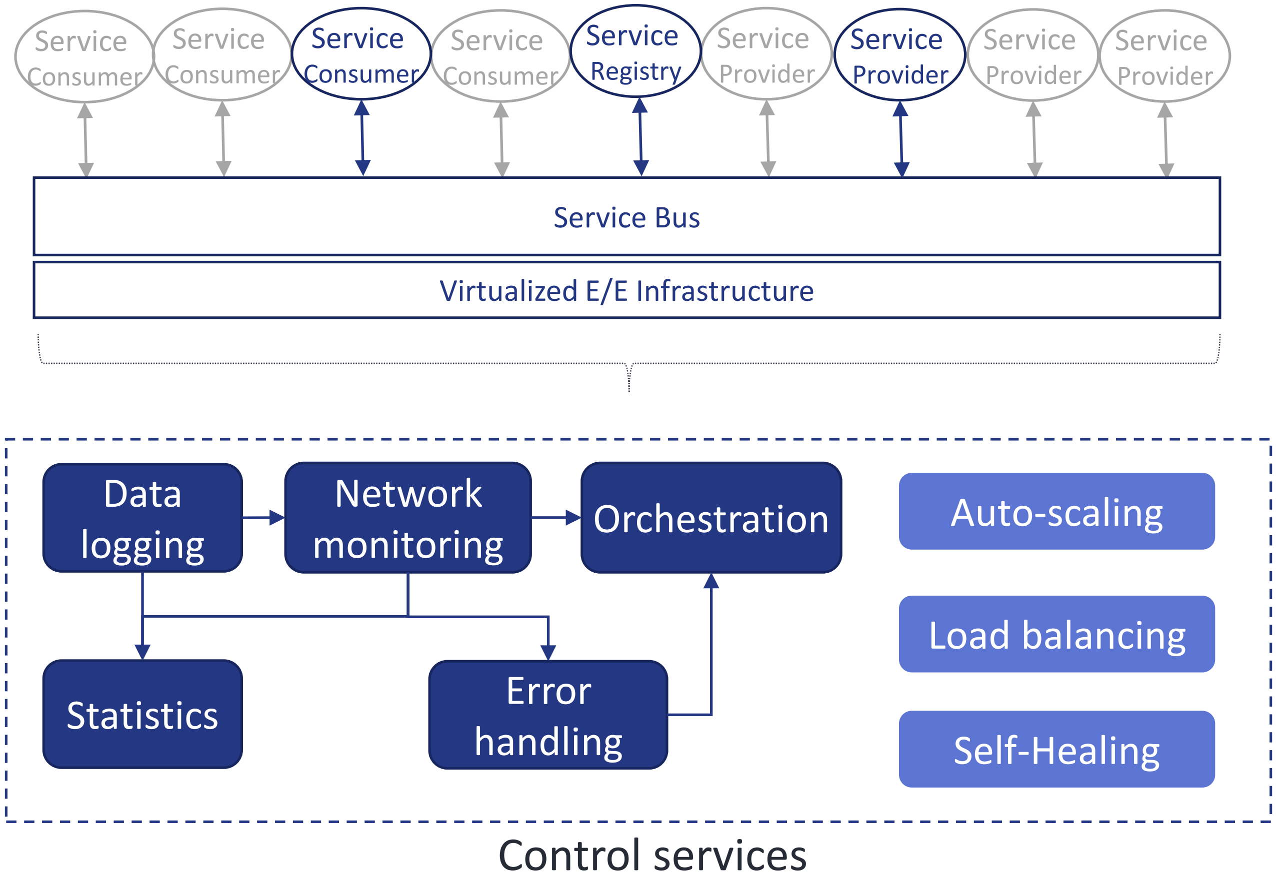
Standardization of protocol deployment:
- Data model definition
- Data version evolution
- Software stack evolution
- Non-functional variant management

SOA architecture style seems to be the solution for reducing the actual complexity

A service is:
- A software service
- Self-contained
- A black box to service consumers
- May be composed of other services
- Independently deployable
- Independently updatable

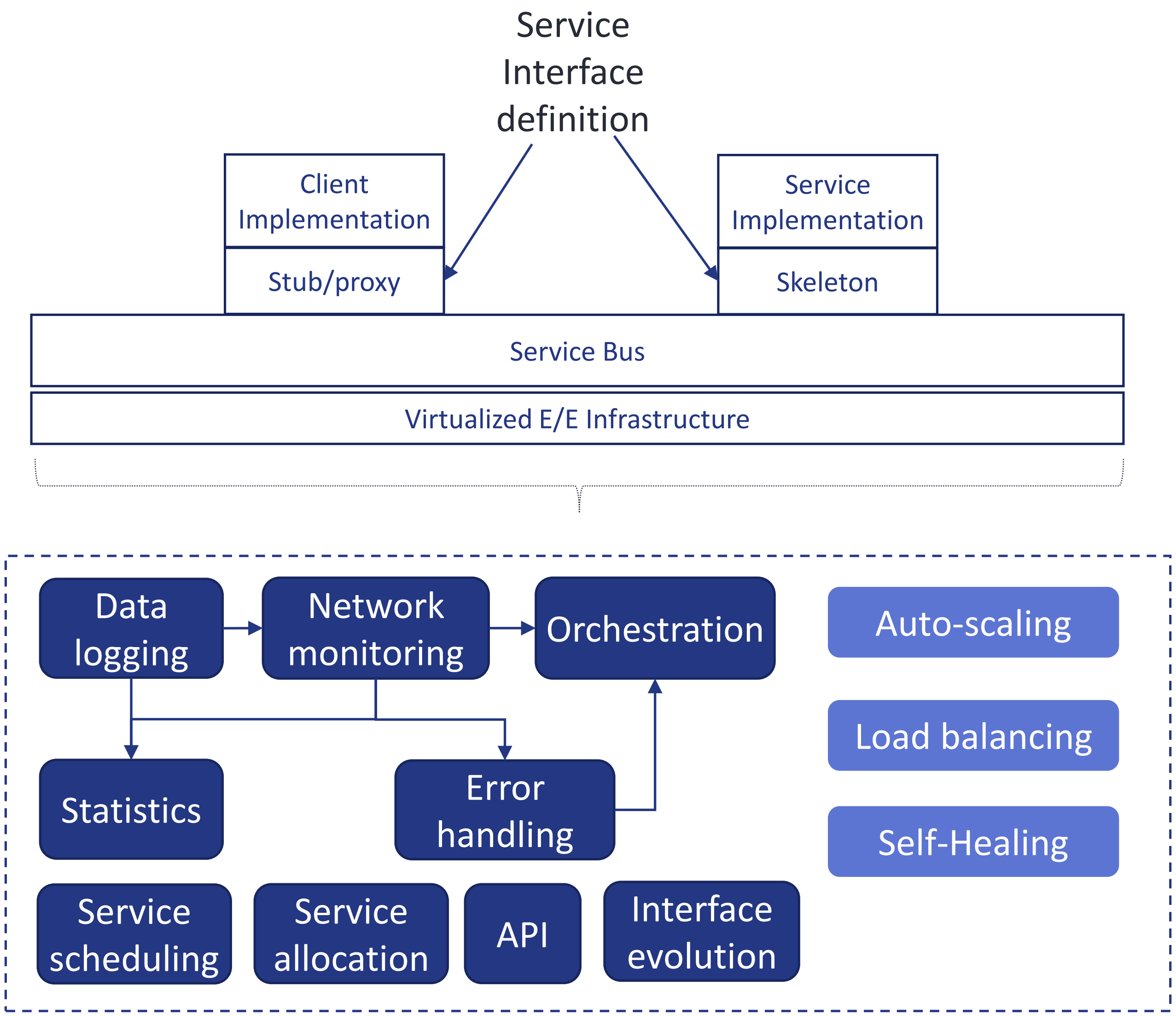


Why communication management?



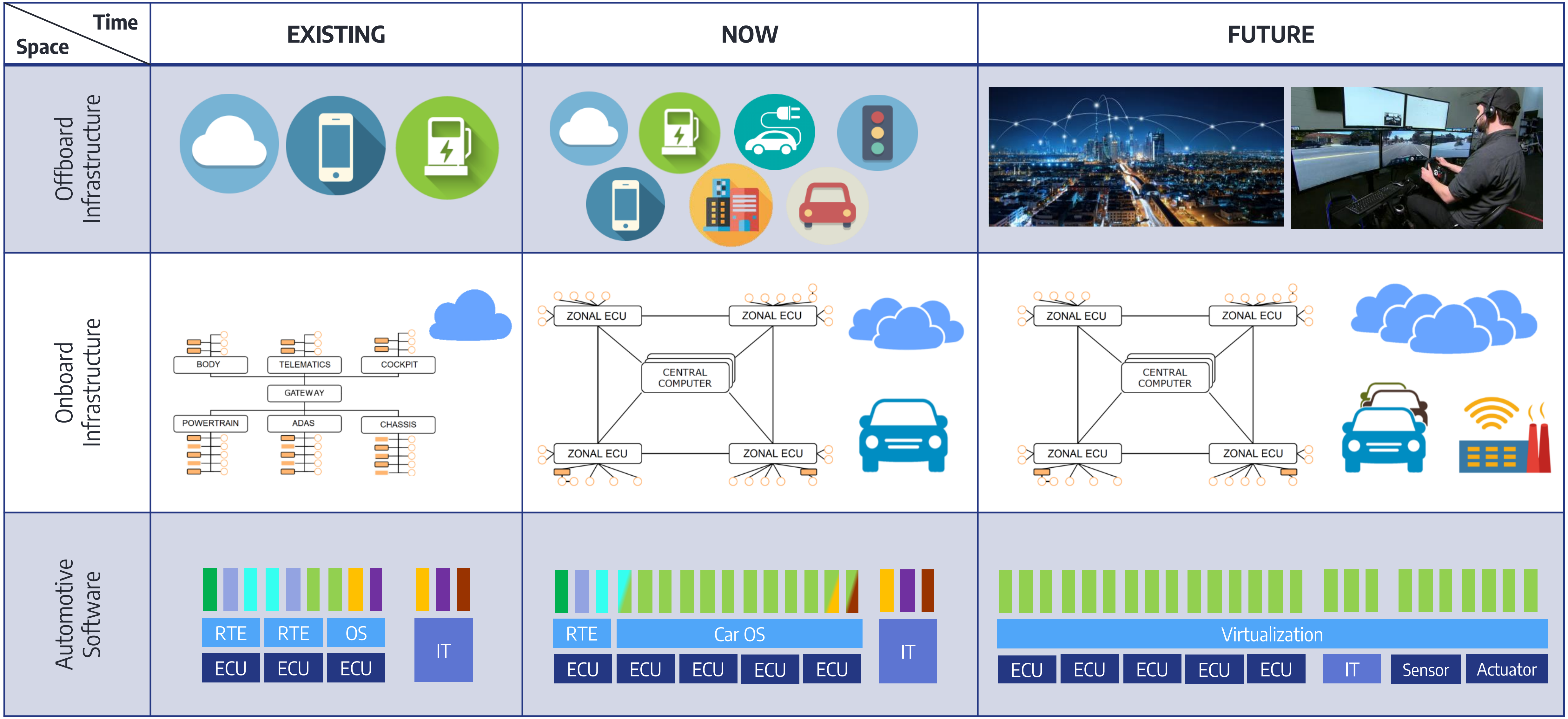
SOA = Software + Network + Business process

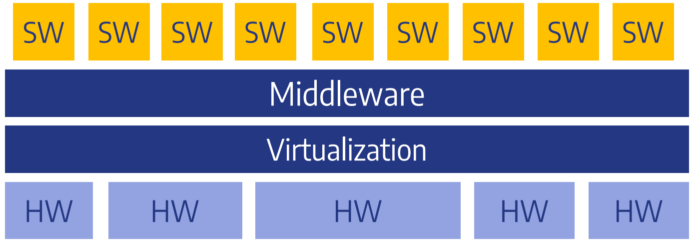
- **Modularity**
  - Application Abstraction
  - Granularity
  - Service Composition
  - Reusability
  - Infrastructure Virtualization
  - Scalability
- **Loose coupling**
  - Fault Tolerance
  - Flexibility
  - Orchestration
- **Open Standards**
  - Standardization
  - Interoperability
- **Simplicity**
  - Reduced Complexity



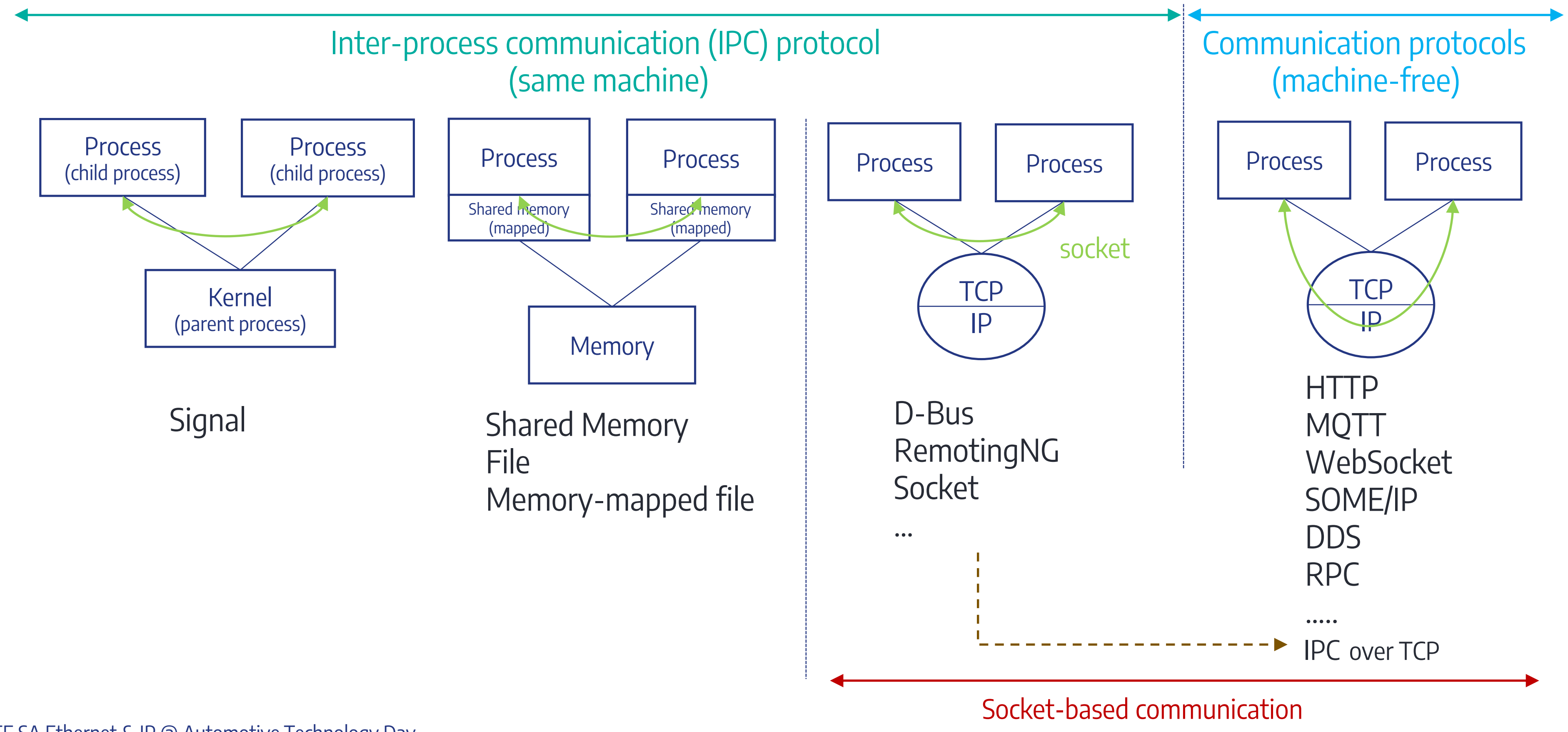
Control services

SOA FOR CAR PLATFORM EVOLUTION

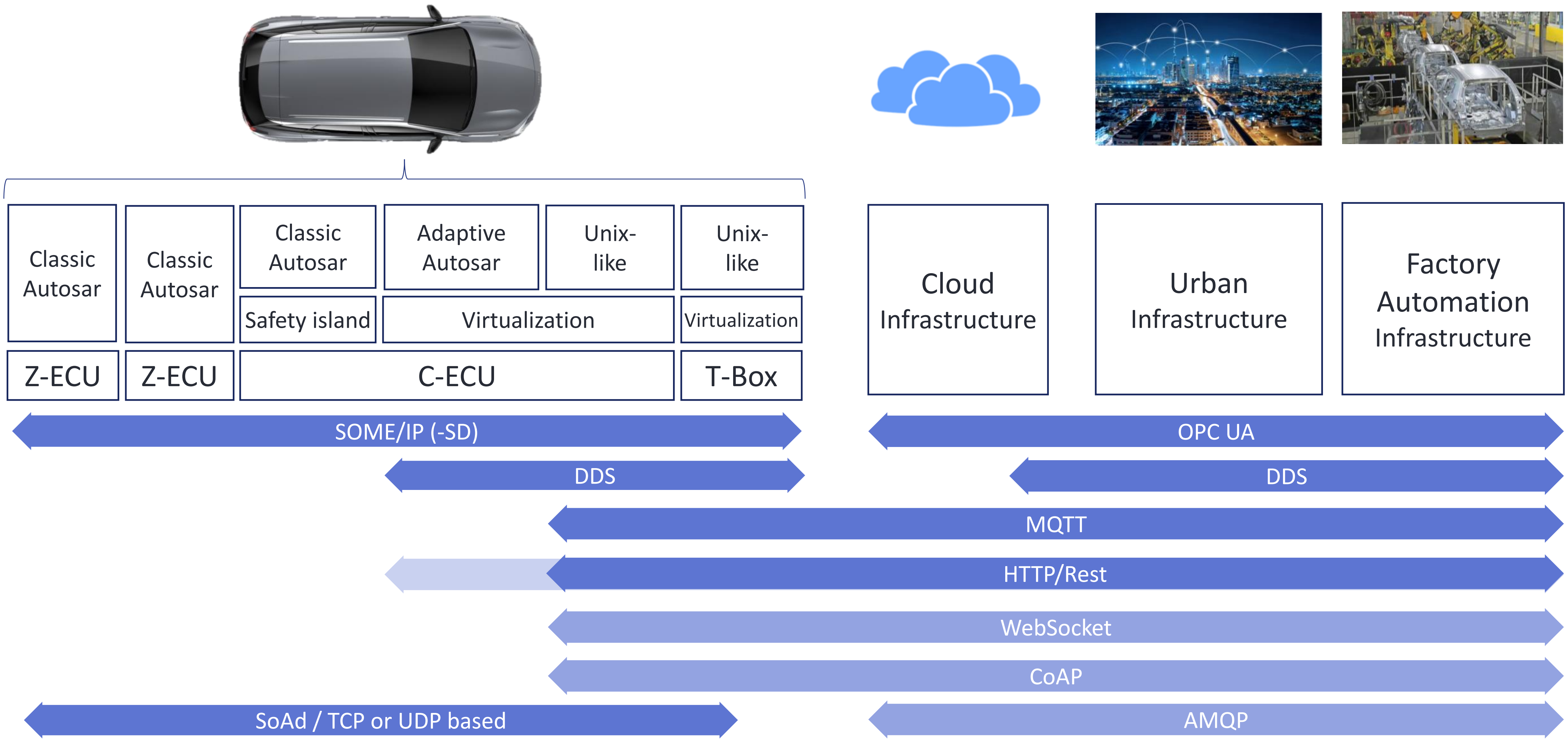




- IPC (inter-process communication)
- Communication protocols



AUTOMOTIVE SOA COMMUNICATION ARCHITECTURE



SOA PROTOCOL COMPARISON

STELLANTIS

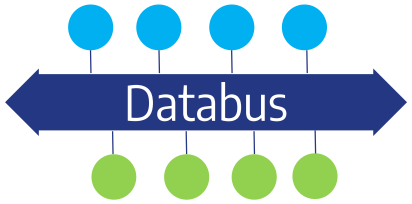
SOA Criteria	AMQP	MQTT	DDS	SOME/IP	OPC UA	CoAP	HTTP/REST	WebSocket
Communication Paradigm	P/S	P/S	P/S	P/S | R/R	P/S | R/R	P/S | R/R	R/R	R/R
Topology	N-1-N or 1-1	N-1-N	N-N	1-N | N-1	N-1 | N-1-N or N-N	N-1 | N-1-N	N-1	N-1
OS Available	L, A	L, A, ~AA	L, A, AA	L, ~A, CA, AA	L, A, ~AA	L, A, ~AA	L, A, ~AA	L, A, ~AA
IDL	-	Franca*	OMG	Franca	OMG	-	REST, Franca*	WEB, Franca*
Discovery	✗	✗	Data	Service	Server	✗	✗	✗
Asynchronous available	✓	✓	✓	✓	✓	✓	✗	✓
Broker-based	✓ | ✗	✓	✗	✗	✓ | ✗	✗	✗	✗
QoS Awareness	Availability	Reliability	✓	✗	✗	✗	✗	✗
TLS / DTLS	TLS	TLS	TLS/DTLS	TLS/DTLS	TLS/DTLS	DTLS	TLS	TLS
Extra Security in the protocol	Session authentication	Session authentication	Session authentication Operation access control	Data authentication Data confidentiality	Session authentication Data confidentiality Operation access control	(Separated study)	(Separated study)	(Separated study) Session authentication
Application	Cloud	Car-Cloud & In-Vehicle	In-Vehicle	In-Vehicle	Car-Cloud	Car-Cloud	Car-Cloud & In-Vehicle	Car-Cloud & In-Vehicle

R/R = Request/Response; P/S = Publish/Subscribe. L = Linux; CA= Classic Autosar; AA= Adaptive Autosar; A= Android.
* Require extra convertor to use FrancaIDL.

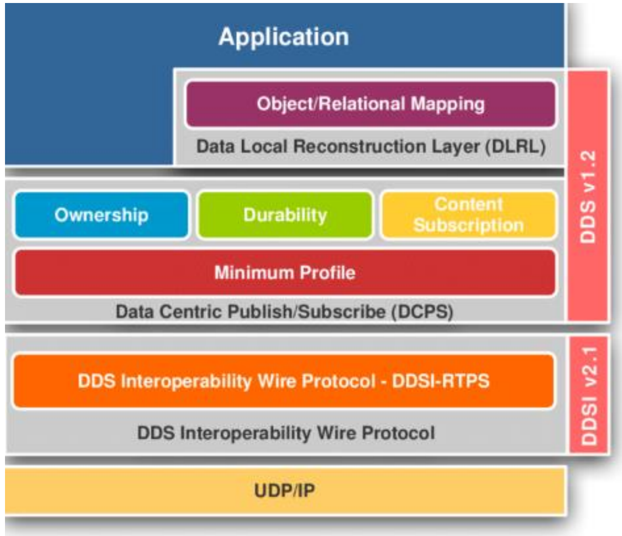
DDS	SOME/IP	OPC UA
Data-centric	Automotive SOA	Interoperability

DDS

Data-centric



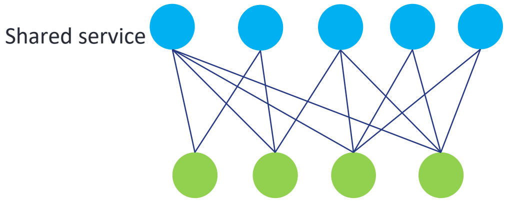
Data addressing
Non-legacy ECU communication
App abstraction thru named network data
Data discovery / Location transparence



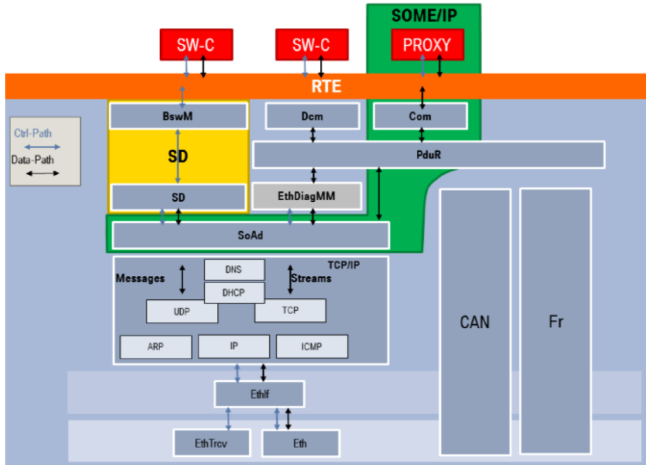
(Source: omg.org/spec/DDS)

SOME/IP

Automotive SOA



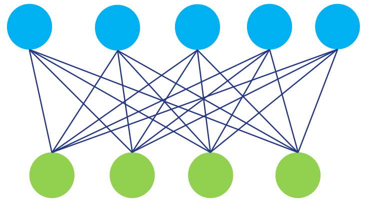
Service interface
Legacy ECU communication
Service discovery



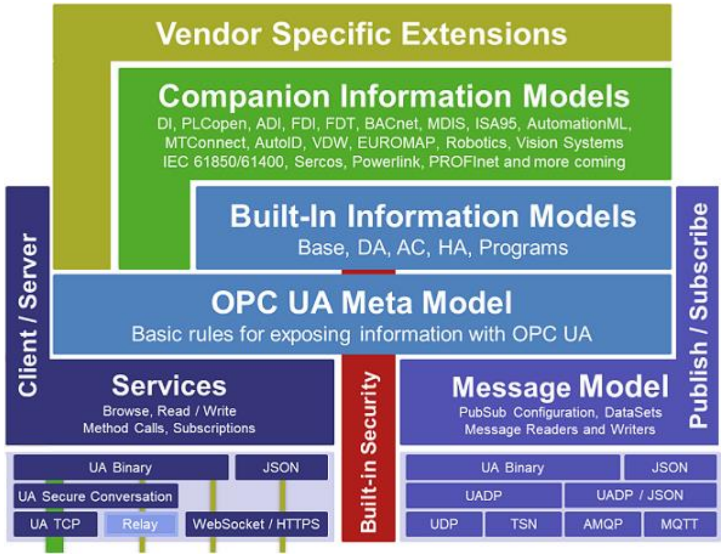
(Source: autosar.org)

OPC UA

Interoperability



Data modeling
Non-legacy ECU communication
Protocol abstraction thru common data models
Server discovery (host/session discovery)



(Source: opcfoundation.org)

Heterogeneity	• Ensure the solution is compatible with the multiple ecosystems existing in the car (Classic and Adaptive Autosar, Android, Linux, etc) • Define and comply with standards and the global deployment pipelines
Dynamicity	• Deploy dynamic services from stateful to stateless in terms of security certifications, capacities, etc. • Reduce service complexity (towards fine-grained services) • Respect the latency limits required by actual and new applications
Development Process	• Develop new interface and network standards and all the control layer services • Evolve the development and testing process of new software requirements • Transform or repackage the legacy applications for new SW contexts
Resource Consumption	• Maintain lightweight consumption for the control-plane services and SOA related mechanisms • Handle simultaneously different types of communication protocols (pub/sub, req/resp…) and different QoS levels
RAMS & Cybersecurity	• Handle compatibility vulnerabilities of onboard/offboard heterogeneous SW contexts. • Rethink and adapt the current failover mechanisms to meet safety requirements • Deploy security mechanisms to prohibit the unauthorized access to the local/remote attack surface

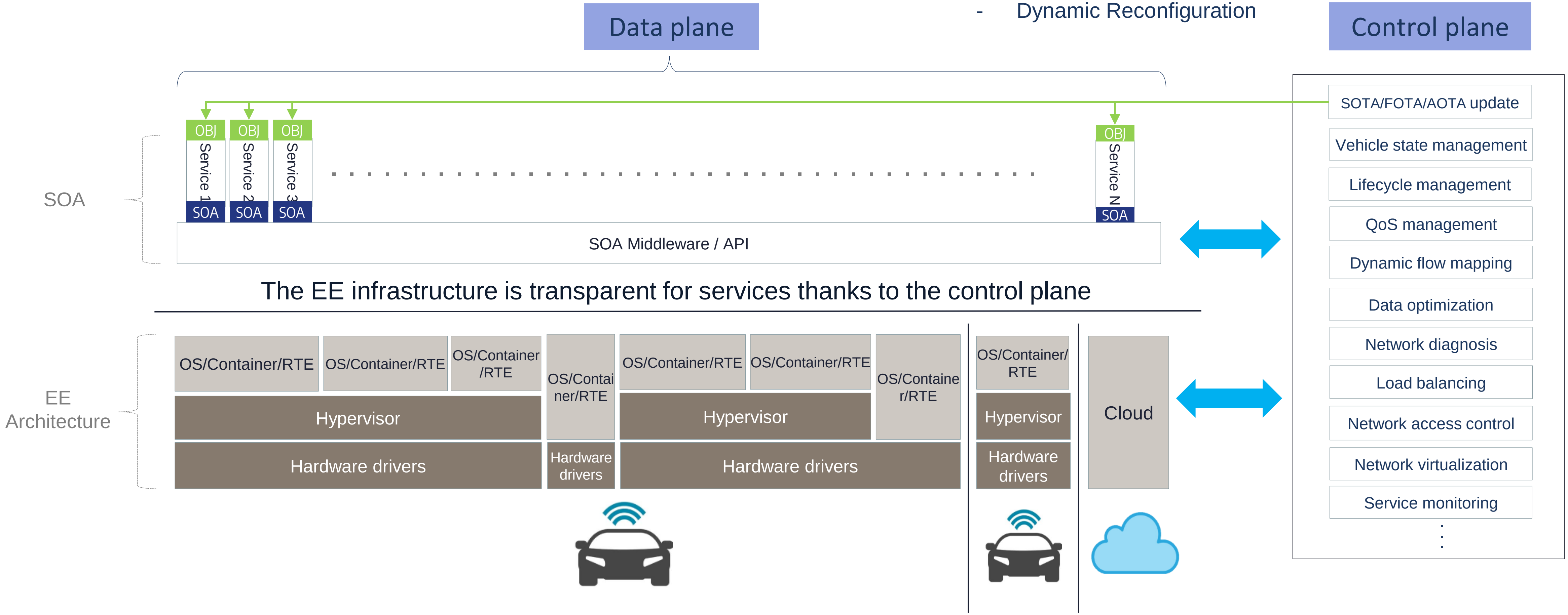
E/E ARCHITECTURE OVERVIEW

SOA COMMUNICATION PROTOCOLS

TAKEAWAYS & PERSPECTIVE

New services in SOA-compatible EE Architecture
Some services may stay in control plane

Orchestration in control plane
- Task Assignment
- Network Supervision
- Error Handling
- Dynamic Reconfiguration



SOA requires the communication management and the service-oriented infrastructure

Standardization is the key to reduce the actual complexity

Different protocols aim at solving different problems, i.e., legacy vs interoperability vs data-centric

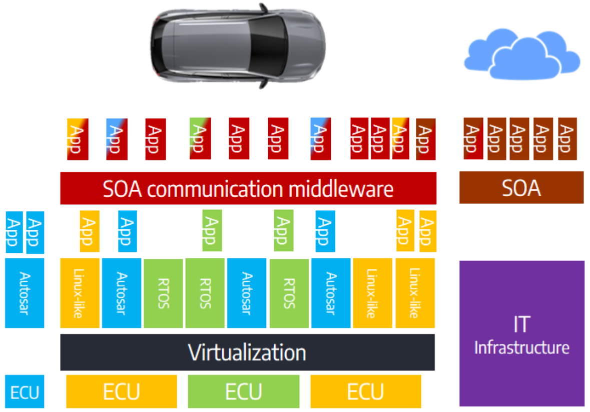
Complicated to unify communication protocols, but easier to unify data/information models

Data abstraction interface is essential for SOA deployment, i.e., API

Both data interface and software components are updatable and require branching and merging strategies

Explicit service design policy for data and control planes, respectively

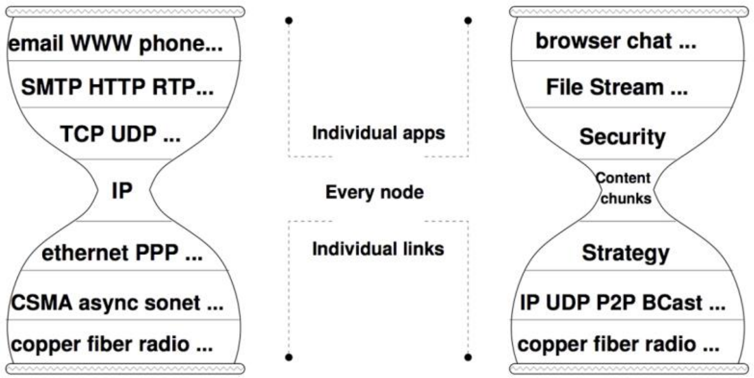
HW/SW decoupling thanks to control plane



STELLANTIS

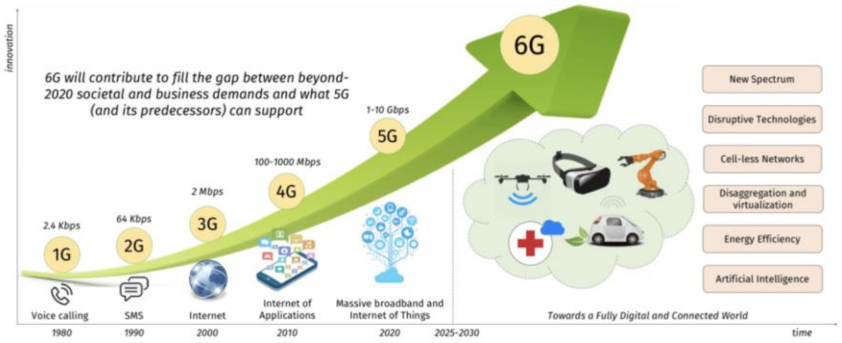
While looking at the evolution of...

From TCP/IP to information-centric network



(Source: named-data.net)

From 5G to 6G



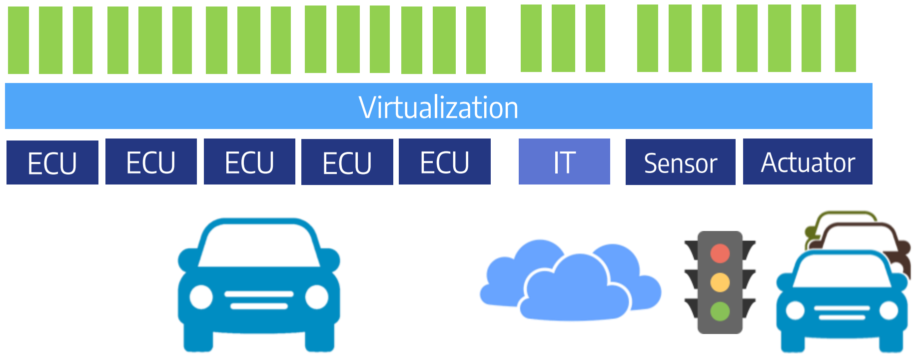
(Source: M. Giordani et al "Toward 6G Networks: Use Cases and Technologies," in IEEE Communications Magazine, vol. 58, no. 3, pp. 55-61, March 2020)

The trend is towards Data-centric

Data modeling
Data addressing, i.e., topic or named data

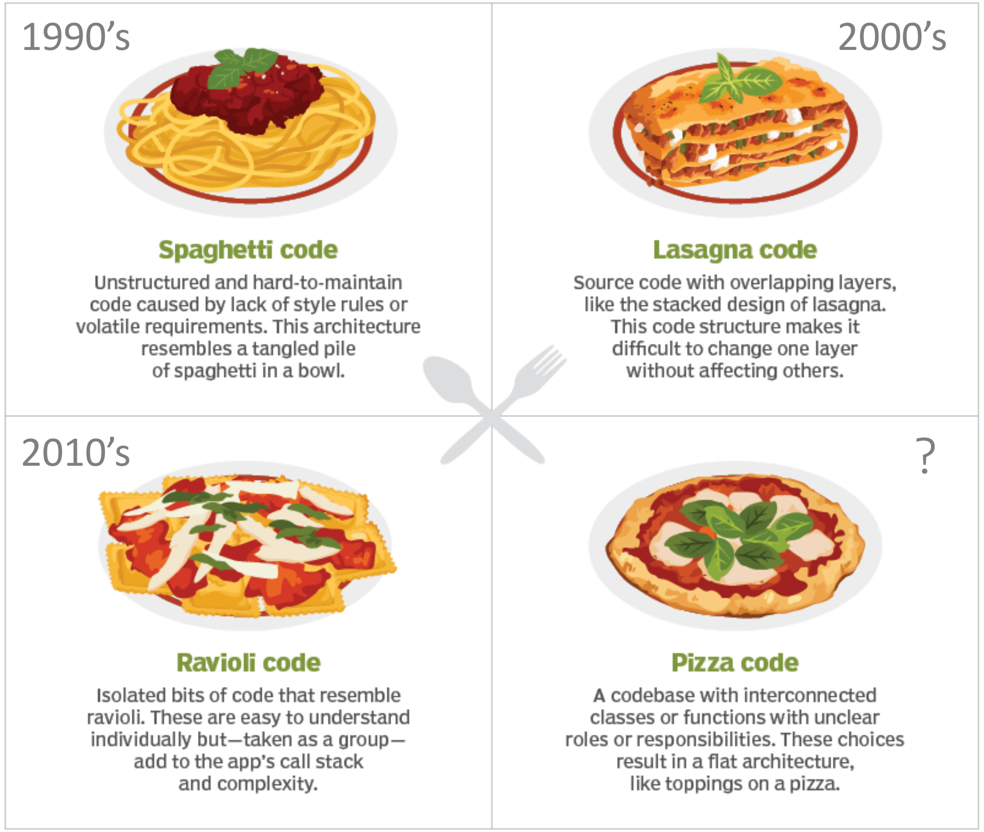
Neither data producer nor data consumer is permanent.
Apps are short-lived, constantly added/updated/removed.

Few programming effort
Scalable software architecture



Q&A TIME

Evolution of Software Architecture



(Source: techtarget.com)