



Adnan Bekan

Senior Research Engineer: Data
Architecture, Connected Vehicle,
Standards & Platforms
BMW Group

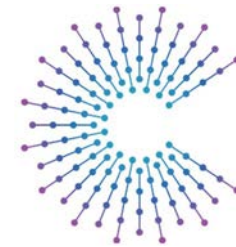
Intro to Vehicle Signal Specification (VSS)



Introduction to Vehicle Signal Specification

ALL MEMBER MEETING

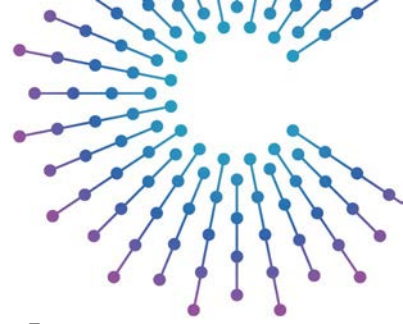
October 10-12, 2023



COVESA

Accelerating the future of connected vehicles

Agenda



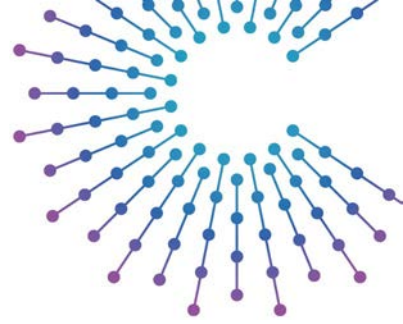
What will be covered:

- What VSS is and what it is NOT.
- The value of VSS.
- Adoption Paths.
- Resources.
- Current Release.
- Next steps.
- Example with service architecture.

What will not be covered:

- Technical deep dive.
- Detailed implementations.

What is VSS ?



[COVESA's Vehicle Signal Specification \(VSS\)](#) is a [common approach](#) for describing vehicle data.

What does it offer: It is a widely adopted; developer friendly, extensible data model & catalog with industry supported tooling. VSS provides a common understanding across the value chain of the connected vehicle.

What value does it bring: This enables improved interoperability and integration which ultimately saves time and cost thus allowing companies to focus on business value creation and differentiating solutions.

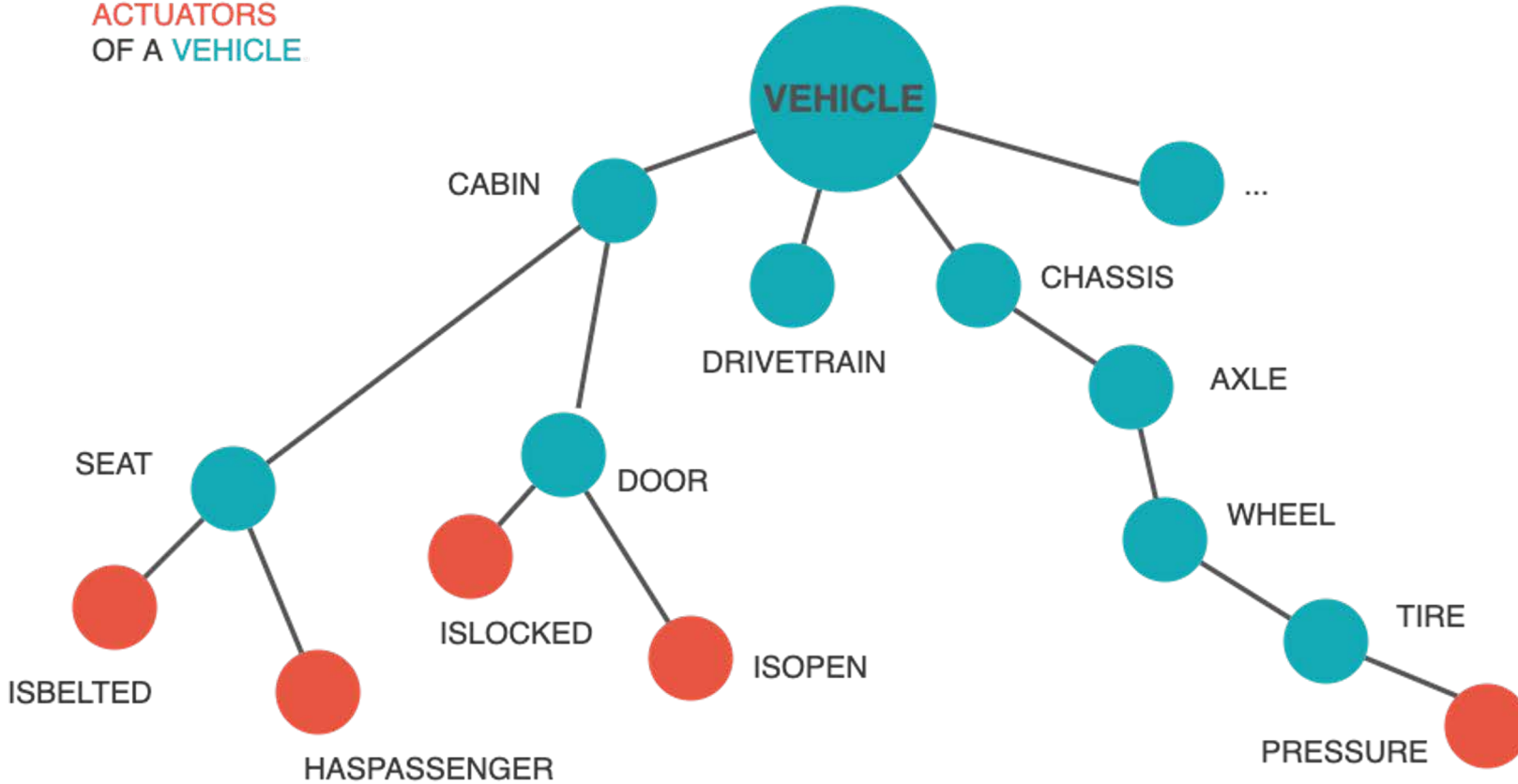
What is VSS - Point by Point

- A simple, flexible and protocol agnostic common approach for describing vehicle data for machines & humans.
- Extensible data model & catalog with industry supported tooling.
- Provides a common understanding across the value chain of the Connected Vehicle.
- Enables improved interoperability and integration, saving time and cost.
- Can be used in car, cloud, edge, or wherever you need it.
- Allows focus on business value creation and differentiating products and solutions.
- Widely adopted.
- Developer friendly.



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



```
Vehicle.Drivetrain.Transmission.Speed  
type: sensor  
datatype: float  
unit: km/h  
description: The vehicle speed as measured by the drivetrain
```

YAML SPECIFICATION



*.csv

*.json

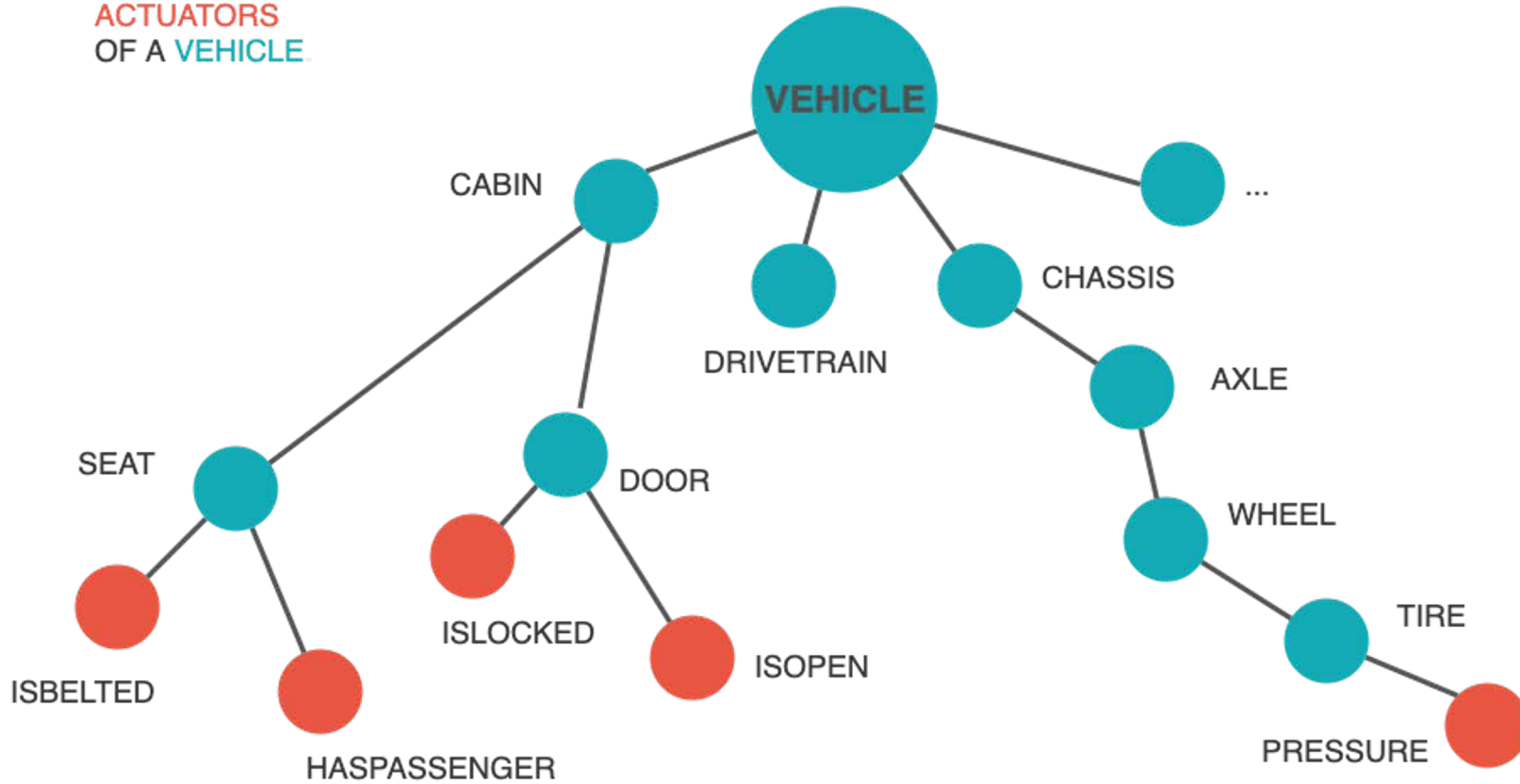
*.graphql

*.so



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



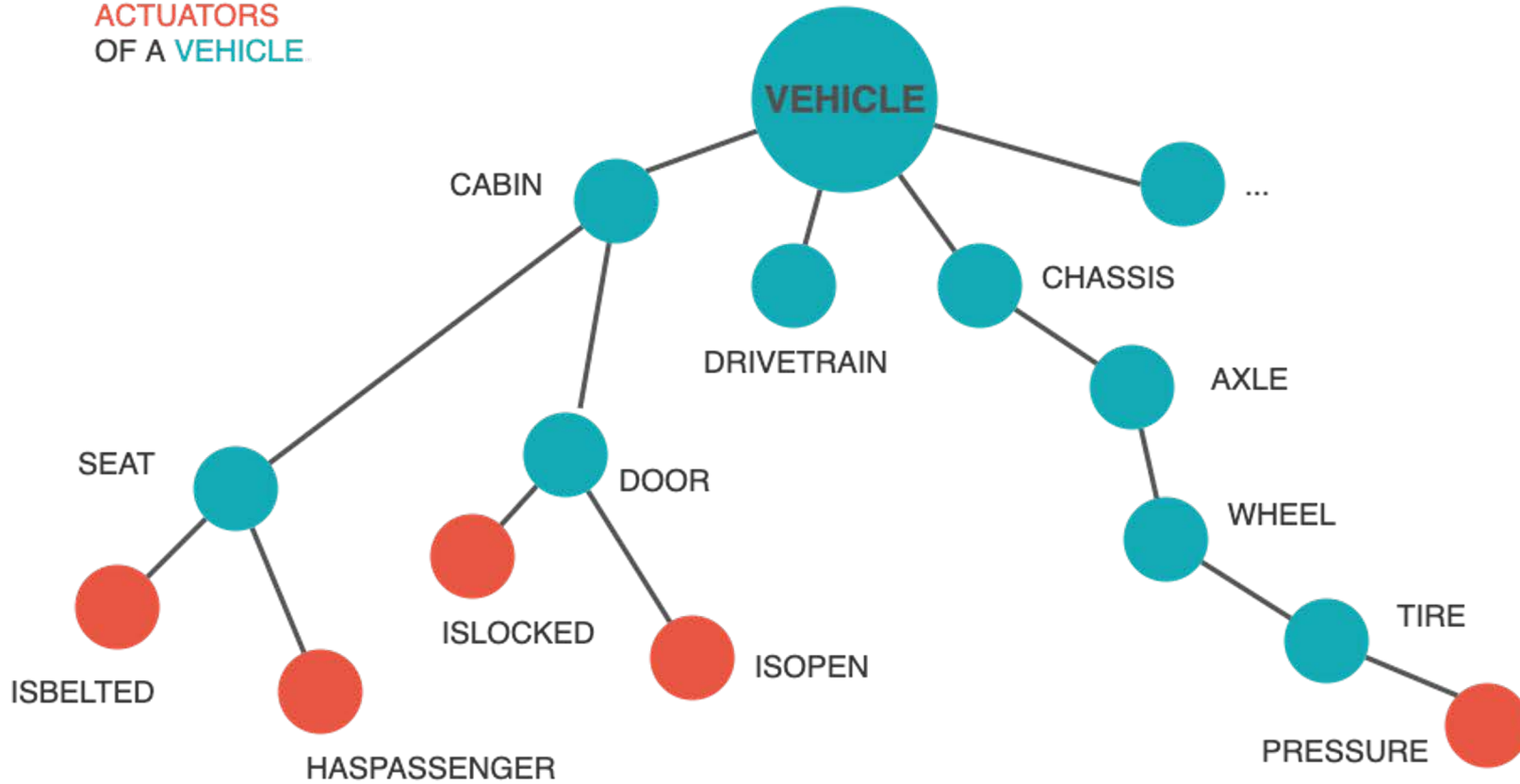
Vehicle.Drivetrain.Transmission.Speed
type: sensor
datatype: float
unit: km/h
description: The vehicle speed as measured by the drivetrain

YAML SPECIFICATION



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



Vehicle.Drivetrain.Transmission.Speed

type: sensor

datatype: float

unit: km/h

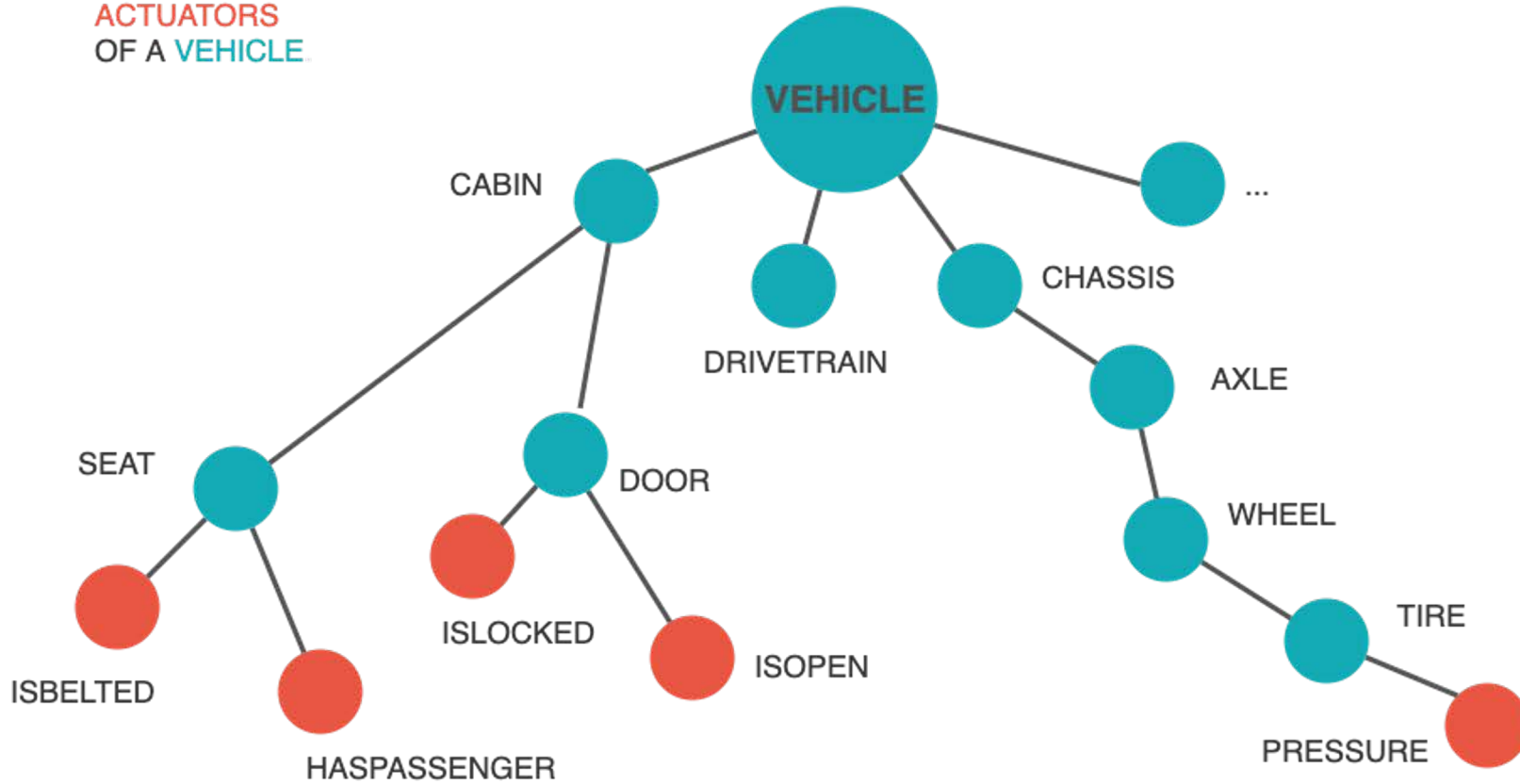
description: The vehicle speed as measured by the drivetrain

YAML SPECIFICATION



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



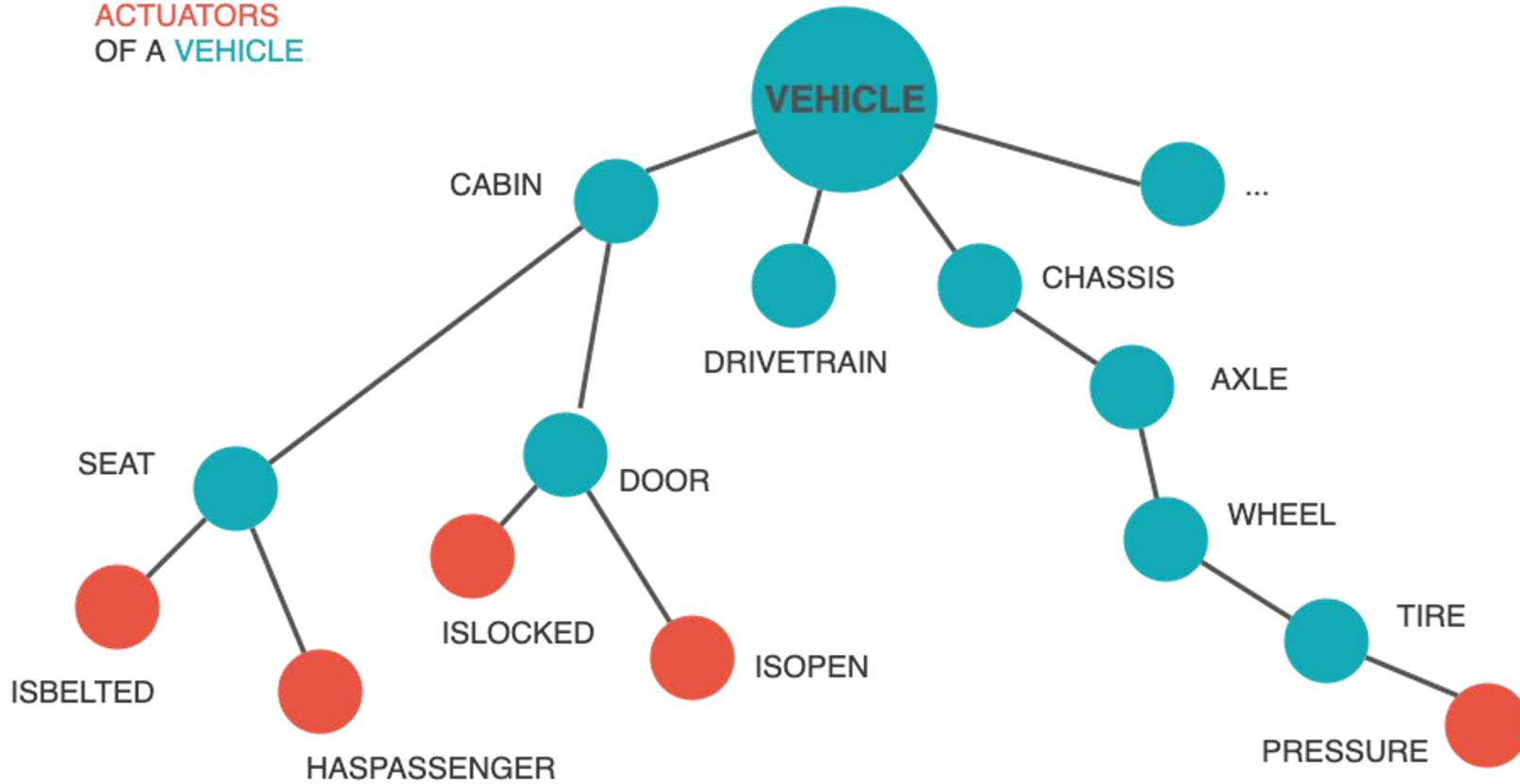
```
Vehicle.Drivetrain.Transmission.Speed  
type: sensor  
datatype: float  
unit: km/h  
description: The vehicle speed as measured by the drivetrain
```

YAML SPECIFICATION



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



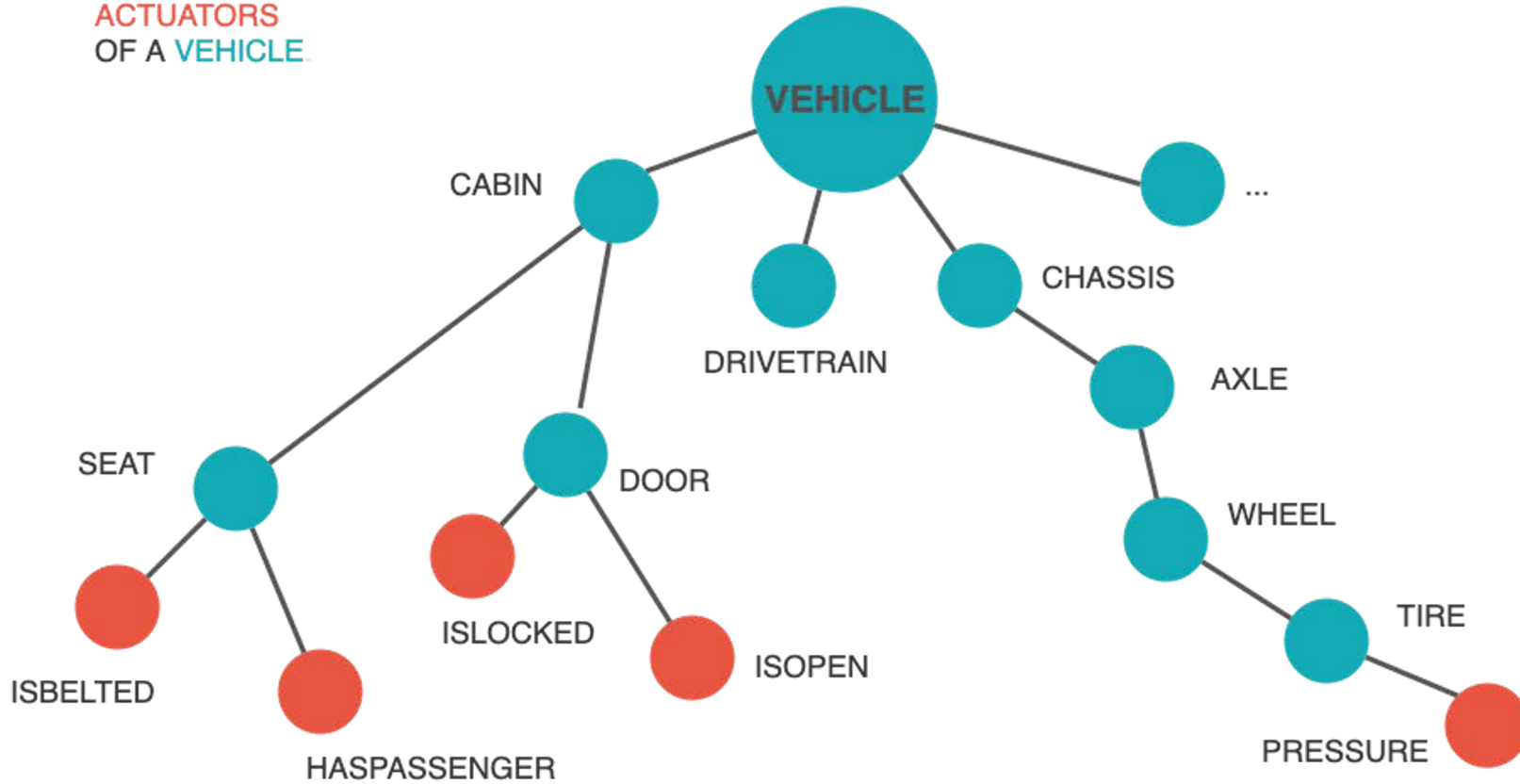
```
Vehicle.Drivetrain.Transmission.Speed  
type: sensor  
datatype: float  
unit: km/h  
description: The vehicle speed as measured by the drivetrain
```

YAML SPECIFICATION



VEHICLE SIGNAL SPECIFICATION

TAXONOMY
FOR
ATTRIBUTES,
SENSORS AND
ACTUATORS
OF A VEHICLE



```
Vehicle.Drivetrain.Transmission.Speed  
type: sensor  
datatype: float  
unit: km/h
```

description: The vehicle speed as measured by the drivetrain

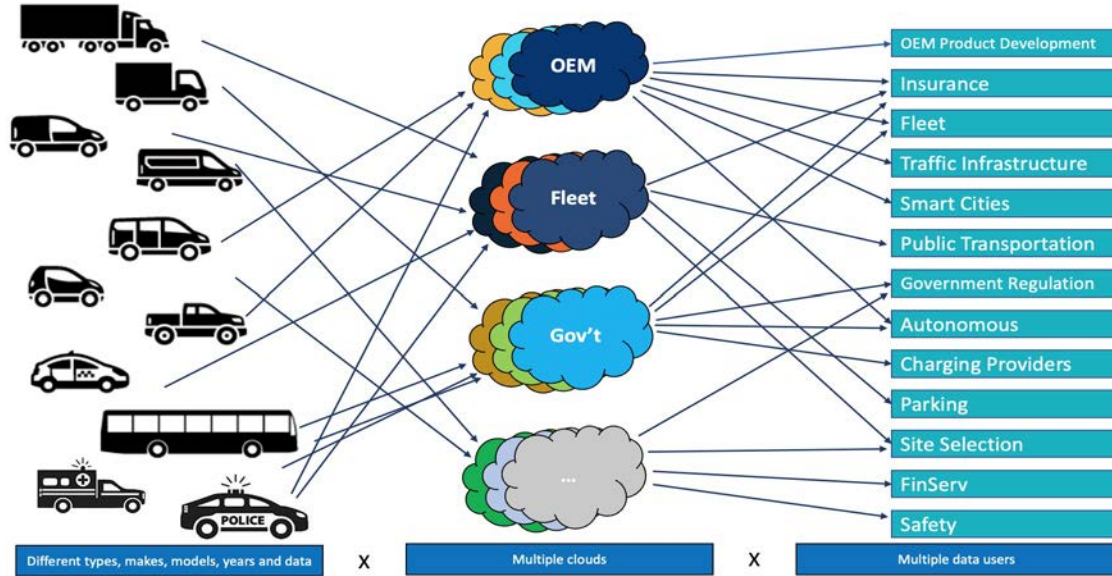
YAML SPECIFICATION

VSS is not...

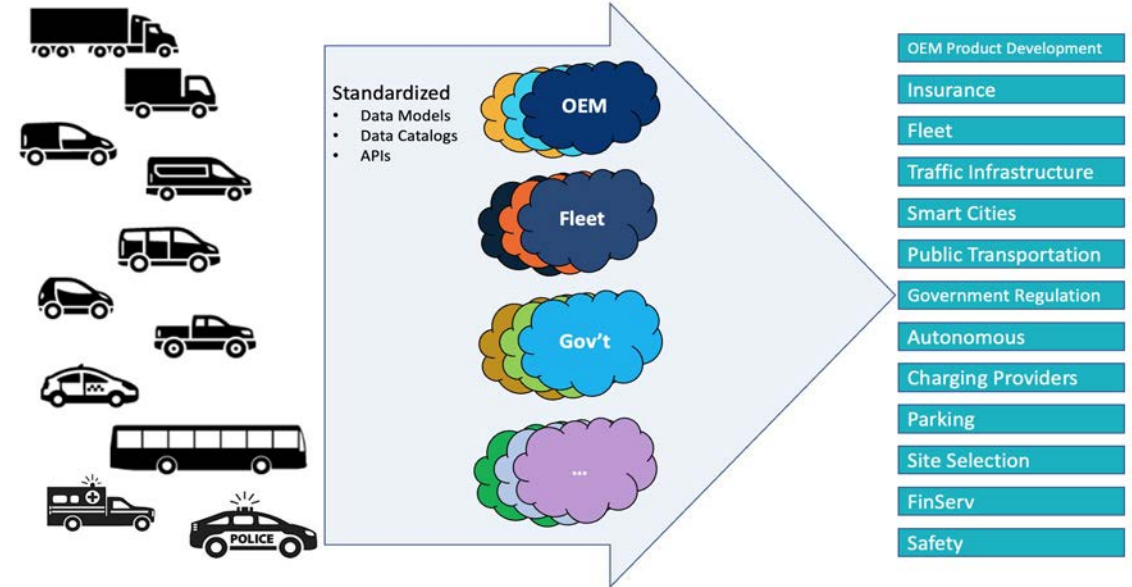
- a protocol
- a serialization format
- an API specification
- Interface Definition Language
- a model for everything in the world
- limited to car, cloud, edge, or...
- a server specification
- an ontology

Why Now?

Today: Integration efforts are complex and unsustainable.



VSS + Consistent Interface reduce complex integration effort and promotes focus on value



Current Reality

- 100 million connected vehicles today
- 400 million connected vehicles by 2025
- Vehicles generate hundreds of GB of data a day or more
- Average vehicle has thousands of signals
- Multiple vehicle models, model years and variation per OEM
- Multiple Clouds (OEM, Fleet, Gov't...)
- Multiple data consumers: OEM, Fleet, Smart City, Insurance, Traffic Infrastructure, Finserv, Charging Providers, ...

Challenges

- Siloed data by organization and domain
- Complex integration and an interoperability
- Estimated 80% of time focused on integration and 20% focused on value
- A lot of noise!

Solutions

- Standardize data models
- Standardize data catalogs
- Standardize APIs

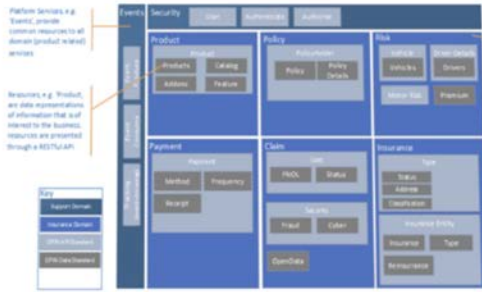
But, where it is used?

- OEM
- Fleet Management
- Usage Based Insurance integrations
- Road Safety
- Traffic Infrastructure and Planning
- Large government projects
- Use by other standards

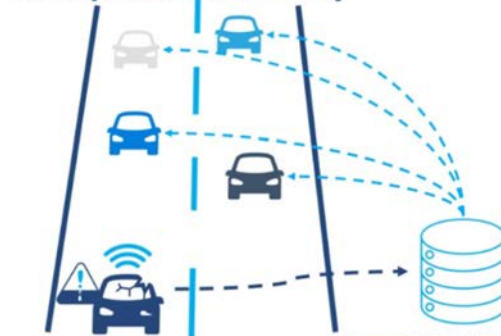
How VSS and Open insurance collaborate to combine open opportunities

Open Insurance Domain Driven Design Model.

- Insurance events for real-time Policy, Claims, Cover and Payments
- Car events for a Vehicle domain based on real time VSS Signals augmenting and driving the Insurance and Underwriting decision
- Real Time exposure to Car cover and Insurance events supported by VSS and Insurance



Initiatives "Data for road safety" and "Vision Zero" to improve road safety



© 2023 Geotab Inc. All Rights Reserved
<https://www.dataforroadsafety.eu/>
<https://visionzeronetwerk.org/>



OPEN STANDARDS

INTEROPERABILITY AND REDUCED DEVELOPMENT COSTS

VEHICLE SIGNAL SPECIFICATION (VSS)

KUKSA Connected Car Ecosystem
Proliferating Open Standards

AWS IoT FleetWise

Easily collect, transform, and transfer vehicle data to the cloud

- Easily access standardized, fleet-wide vehicle data**
Build virtual representations of vehicles and apply a common data format to structure and label vehicle attributes, sensors, and signals
- Reduce costs with intelligent data filtering**
Select which data to transfer and define rules and events for when to transfer it, and dynamic data selection automatically reduces redundant data
- Detect and mitigate problems more quickly by surfacing vehicle data in near real time**
Take quick, corrective action by notifying operations or manufacturing

A Mutual Ecosystem of Partners

 PROJECT ROLE Enabling the intelligent and natural interface to everything in the vehicle, driven by the responsive Assistant Platform and ANGO's state of the art conversational AI technology.	 PROJECT ROLE Focusing on exploring improvements in end-user experiences to impact Safety, Personalization, Comfort and open up new Monetization opportunities for our customers.	 PROJECT ROLE Providing a secure, global ecosystem for accurate vehicle telematics data and context integration, within every vehicle and marketplace for connected car mobility services.	 PROJECT ROLE The Automotive Parts Manufacturer's Association (APMA) of Canada is building the first, original, full-build, zero-emission concept vehicle named Project Arrow. This is an all-Canadian effort which is designed, engineered, and built through the joint efforts of our world-class automotive supply vendor and joint-venturer partners.
 PROJECT ROLE Advanced computing and displays for innovative smart cockpit, intelligent vehicle interfaces, using open operating systems fit from concept to mass production.	 PROJECT ROLE Enabling learning and personalization through vehicle data and artificial intelligence designed to keep drivers safe and comfortable.		

Adopters



AWS IoT FleetWise



Relationship to Other Initiatives

W3C Vehicle Information Service Specification (VISS)	VSS is the data model used by W3C VISS
Vehicle Signal Specification Ontology	VSSo is directly correlated with the VSS catalog
Common Vehicle Interfaces	VSS will be a data model used in API definition of services.
Data Architecture	VSS as data model for Vehicle Domain.

Paths to Adoption

Build your own

- Use common catalog to “get started”
- Extend your own catalog privately
- Create a custom catalog

Buy/Use a solution

- Use existing VSS data server
- Buy something VSS-ready...

Contribute & grow

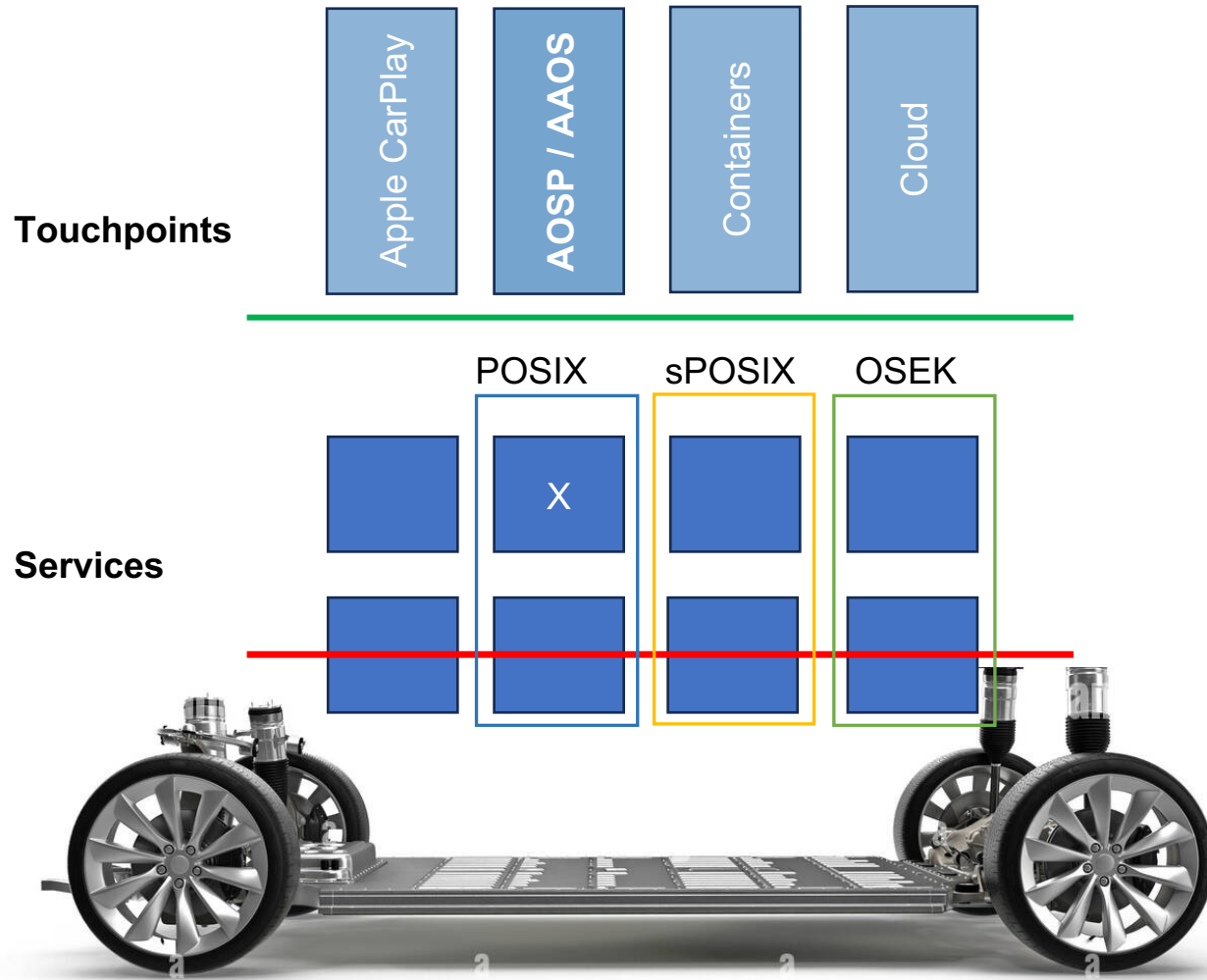
- Become part of the community and help to shape VSS for what you need

In-cloud ←-----→ In-Vehicle

Resources

Documentation	https://covesa.github.io/vehicle_signal_specification/ https://wiki.covesa.global/display/WIK4/VSS+Resources+at+a+Glance	Please read the documentation first. It answers most questions and points you to where to find other information.
Specification	https://github.com/COVESA/vehicle_signal_specification	To see specific nodes such as Cabin or Powertrain drill down into https://github.com/COVESA/vehicle_signal_specification/tree/master/spec
Tools	https://github.com/COVESA/vss-tools	Tools for building and processing VSS
GitHub Issues and Wiki	https://github.com/COVESA/vehicle_signal_specification/issues https://github.com/COVESA/vehicle_signal_specification/wiki	The team uses GitHub Issues and Wiki heavily
Vehicle Signal Specification ontology	https://www.w3.org/TR/vsso/	Ontology built with VSS
Vehicle Information Service Specification	https://www.w3.org/TR/viss2-core/#:~:text=The%20Vehicle%20Information%20Service%20Specification,Vehicle%20Signal%20Specification%20(VSS).	Server specification for accessing vehicle information represented by VSS

Example 1/5 – HLA

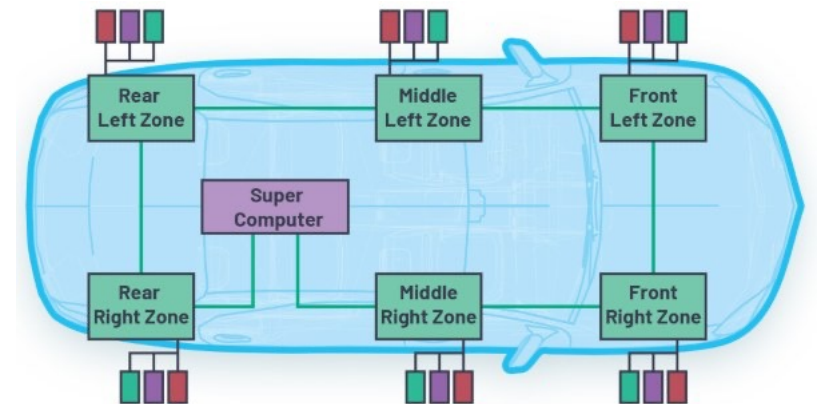


What type of arch?

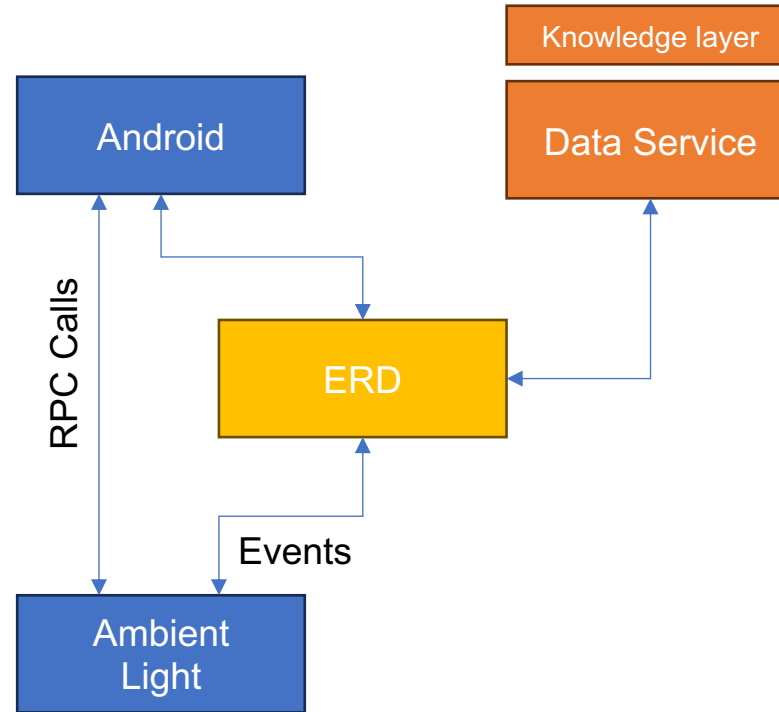
- Client-Server
- Event-Driven

....

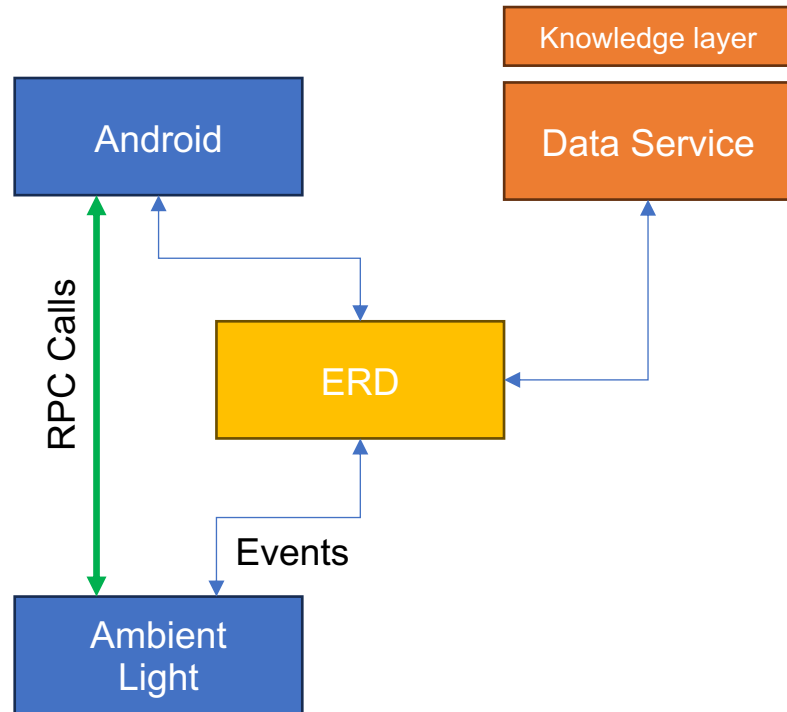
Most common challenges are
Fragmentation, Usability, Integration
(Test and Validation)



Example 2/5 -HLA



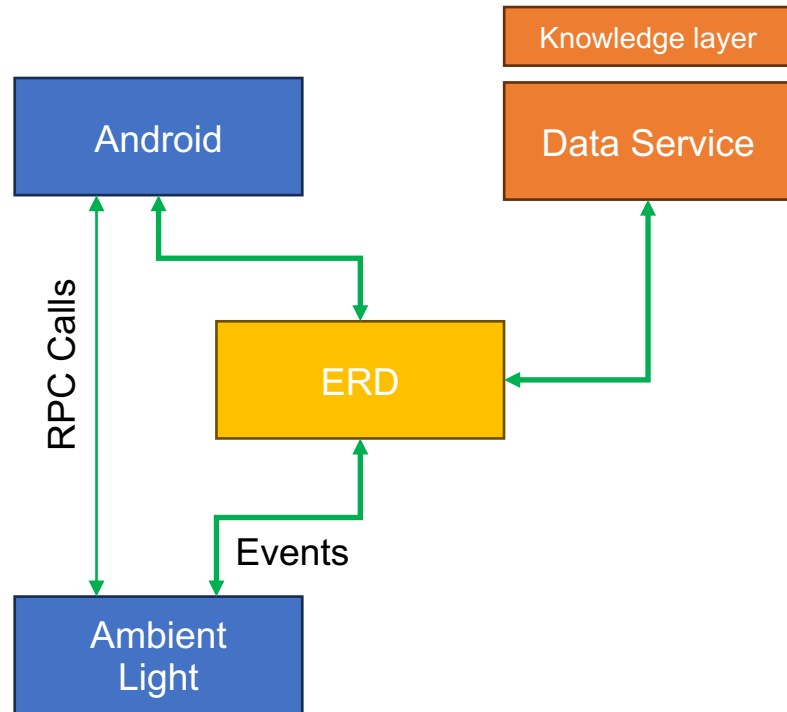
Example 3/5 – “RPC”



```
1 openapi: 3.0.0
2 info:
3   title: Vehicle Cabin Ambient Light API
4   version: 1.0.0
5 paths:
6   /Vehicle/Cabin/InteriorLights/AmbientLight/Segments:
7     get:
8       summary: Get the list of segments
9       responses:
10        200:
11          description: List of segments retrieved successfully
12          content:
13            application/json:
14              schema:
15                type: array
16                items:
17                  $ref: '#/components/schemas/Segment'
18   /Vehicle/Cabin/InteriorLights/AmbientLight/setSegmentsPermanentColor:
19     post:
20       summary: Set permanent color for segments
21       requestBody:
22         required: true
23         content:
24           application/json:
25             schema:
26               type: object
27               properties:
28                 segments:
29                   type: array
30                   items:
31                     $ref: '#/components/schemas/Segment' # Reference to the Segment object
32                 Fading:
33                   type: string
34                   format: duration
35                 ColorSpace:
36                   type: string
37                   enum: [sRgb888Gamma22D65]
38       responses:
39        200:
40          description: Color set successfully
41        400:
42          description: Bad request
43
44   /Vehicle/Cabin/InteriorLights/AmbientLight/segments/{segmentId}/lamps:
45     get:
```

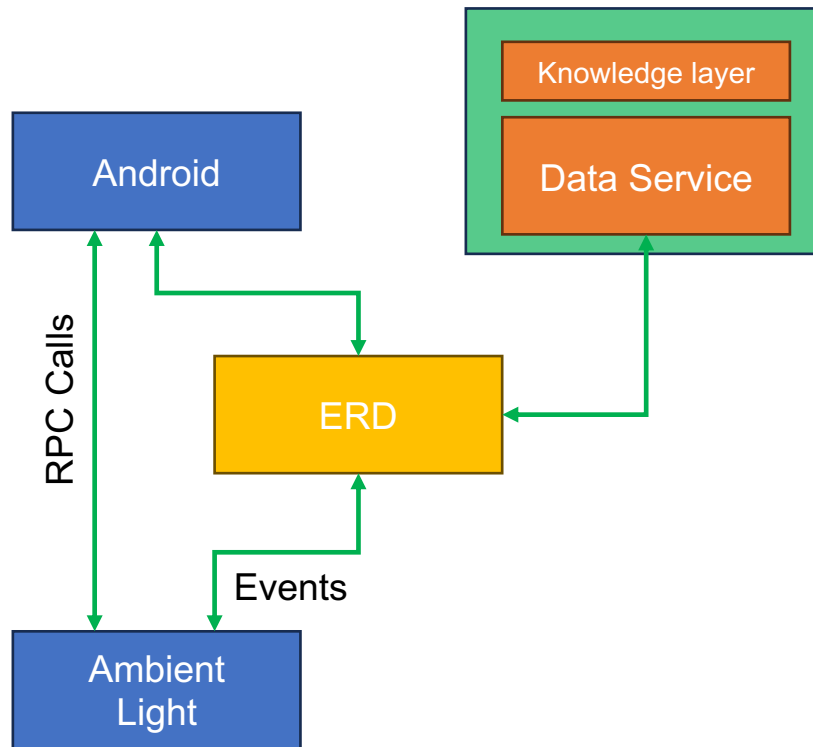
Use generator to convert to protobuf.

Example 4/5 – Events



```
1  asyncapi: '2.6.0'
2  √ info:
3    · title: Vehicle Cabin Ambient Light Notifications
4    · version: '1.0.0'
5  √ description: |
6    This service provides notifications related to ambient lighting in a vehicle
7    cabin.
8  √ channels:
9  √ Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Left/Color:
10   · description: The topic for color change notifications on the left side of Row1.
11   √ publish:
12     · summary: Color Change Notification for Row1 Left
13     · operationId: publishColorChangeForLeft
14   √ message:
15     · $ref: '#/components/messages/Color'
16
17  √ Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Right/Color:
18   · description: The topic for color change notifications on the right side of Row1.
19   √ publish:
20     · summary: Color Change Notification for Row1 Right
21     · operationId: publishColorChangeForRight
22   √ message:
23     · $ref: '#/components/messages/Color'
24
25  √ Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Left/Status:
26   · description: The topic for status change notifications on the left side of Row1.
27   √ publish:
```

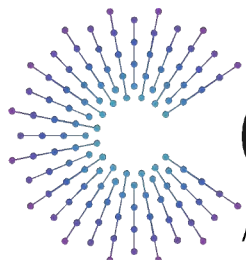
Example 5/5 -HLA



```
{
  "_id": "Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Left/Color",
  "description": "The topic for color change notifications on the left side",
  "summary": "Color Change Notification for Row1 Left",
  "operationId": "publishColorChangeForLeft"
}
{
  "_id": "Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Right/Color",
  "description": "The topic for color change notifications on the right side",
  "summary": "Color Change Notification for Row1 Right",
  "operationId": "publishColorChangeForRight"
}
{
  "_id": "Color",
  "summary": "Notification of color change for Segment Row1",
  "payload": {
    "type": "object",
    "properties": {
      "color": {
        "type": "object",
        "properties": {
          "red": {
            "type": "integer",
            "minimum": 0,
            "maximum": 255
          },
          "green": {
            "type": "integer",
            "minimum": 0,
            "maximum": 255
          },
          "blue": {
            "type": "integer",
            "minimum": 0,
            "maximum": 255
          }
        }
      }
    }
  }
}
{
  "_id": ObjectId("5ff5b6eae1f913a25c000001"),
  "timestamp": ISODate("2023-10-15T12:30:45Z"),
  "channelId": "Vehicle/Cabin/InteriorLights/AmbientLight/Segments/Row1/Left/Color",
  "value": {
    "color": {
      "red": 150,
      "green": 50,
      "blue": 50
    }
  }
}
```


A decorative graphic at the top of the slide consists of a network of interconnected nodes and lines. The nodes are represented by small circles, and the lines are thin, connecting the nodes in a complex, web-like pattern. The colors of the nodes and lines transition from a light blue on the left to a darker blue on the right.

Questions?



COVESA

Accelerating the future of connected vehicles

