Toward a Common Vehicle Data Model

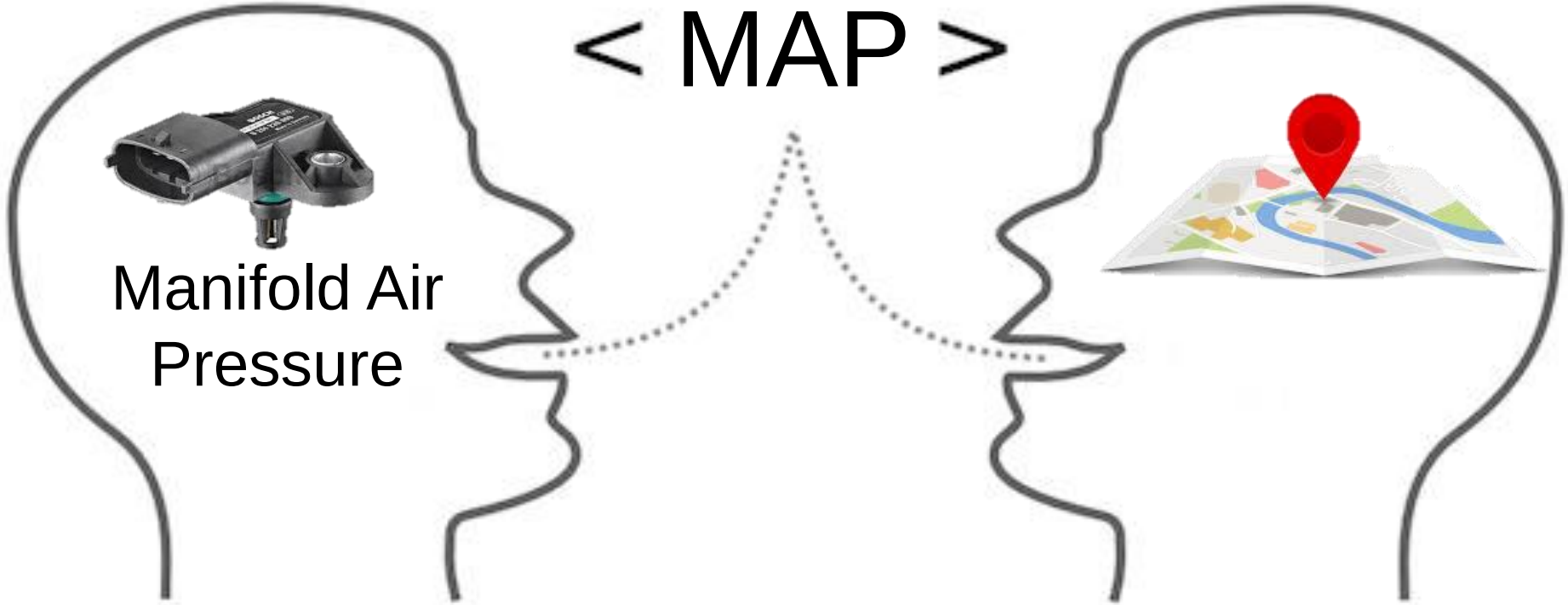
November 16, 2019 | Cloud & Connected Services Workshop Session One

Benjamin Klotz, Gunnar Andersson

Toward a Common Vehicle Data Model

Motivation

Motivation



Heterogeneous vehicle data

{"acceleratorPedal":{"position":"4095","ecoPosition":"3"},"brakeContact":"1
6","speedActual":"0“}, "timeStamp":"2018-01-10T17:01:27.297Z",}

{"name":"accelerator_pedal_position","value":0,"timestamp":1361454211.483000}
{"name":"fuel_level","value":23.478279,"timestamp":1361454211.485000}
{"name":"torque_at_transmission","value":1,"timestamp":1361454211.488000}

Signal name?
Units?
Timestamp?

Temperature sensor

Adaptive cruise control

Front camera

Radar

Blind spot detection

Wheel speed sensor

Oil temperature sensor

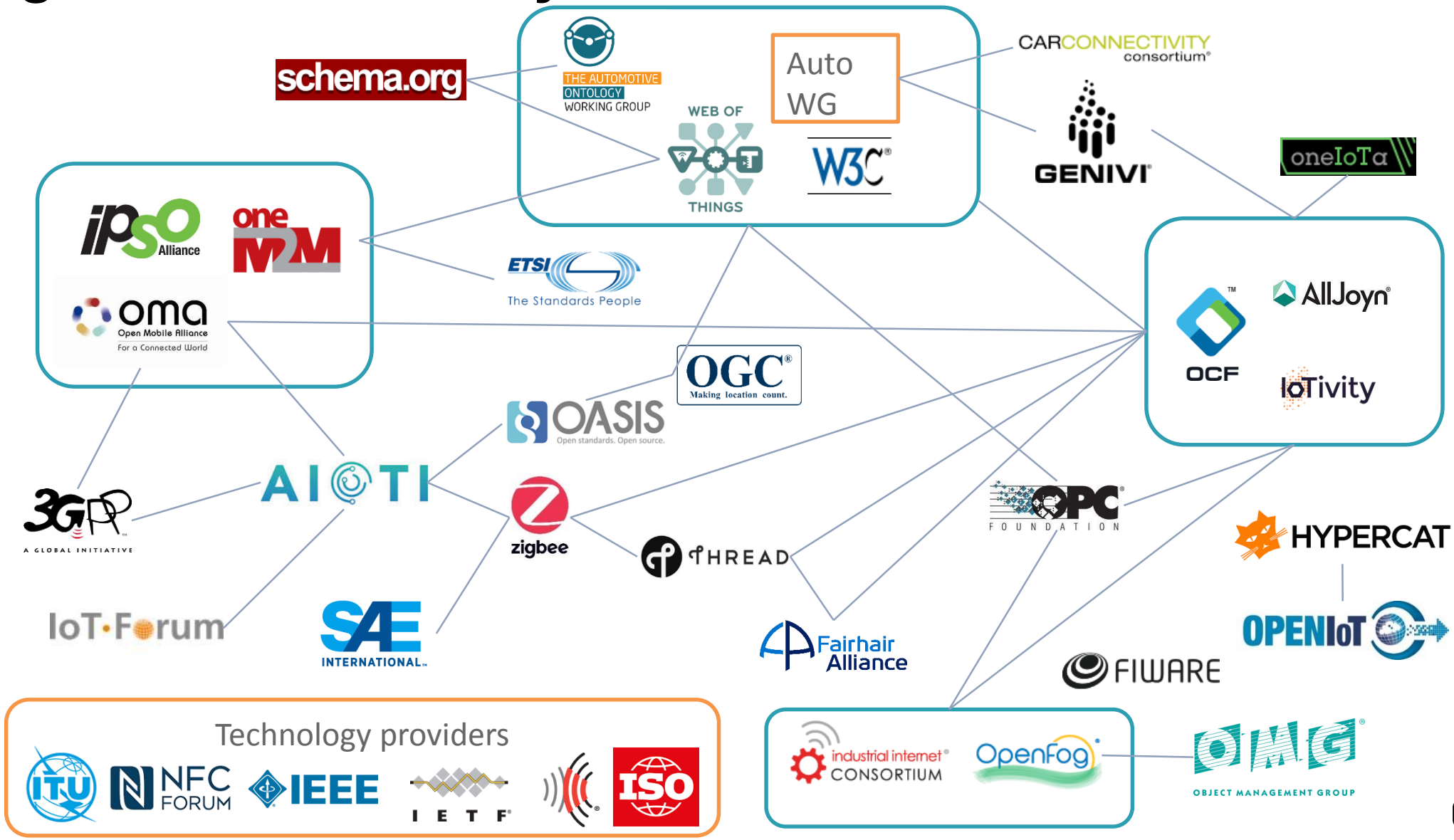
Tire pressure sensor

Steering angle sensor

Park assistant

Vehicle height sensor

Fragmented IoT ecosystem



GENIVI®

Challenges

- Heterogeneous data
  - Sources: vehicles, road infrastructure, external APIs…
  - Different brands and models
- Hard to standardize bus signals (OBDII)
- Access control independence (data model vs data instances)
  - Security
  - Privacy
  - Different implementations
- In-vehicle signals vs backend APIs

GENIVI®

Toward a Common Vehicle Data Model

Gap analysis

GENIVI®

Some major standardization initiatives

- ISO 20078 Extended Vehicle
- W3C Vehicle Information Server
- SensorIS
- AutoMat Common Vehicle Information Model
- Car Connectivity Consortium
- IoT initiatives…

GENIVI®

ISO Extended Vehicle (ISO 20078)

Motivation

- Increasing demand from 3rd parties to access vehicle data and functionality
- OEMs already equipped vehicles with telematics units and IT-infrastructure to handle connectivity
- Need to define a design and requirements to ensure that security, safety and data privacy (best practices, common methods)

Data model

- For 3rd parties to implement
- RESTful with JSON or XML schema
- requirements on several aspects:
  - URI definition,
  - error handling,
  - Naming
  - interaction pattern

Stakeholders

Metadata

Policies: requirrements for 3rd parties

SensorIS

Motivation

- Enable broad access, delivery and processing of vehicle sensor data
- Enable easy exchange of vehicle sensor data between all players
- Enable enriched location-based services
- Drive global growth in this field

Data model

- Data messages in categories (which you can create)
- Identifies of submitter, session, message, vehicle fleet, vehicle, and driver
- Developed in google Protocolbuf library

Stakeholders

Metadata

- Units explicitly defined (e.g. deg_c for Celsius degrees)
- Policy for category extension to be compatible

W3C Vehicle Information Server

Motivation

Data model

- Vehicle Signal Specification as the per default model
- Alternative possible

Stakeholders

Metadata

- Vehicle Signal Specification extension mechanism

Toward a Common Vehicle Data Model

Vehicle Signal Specification @ GENIVI

Vehicle signal specification (VSS)

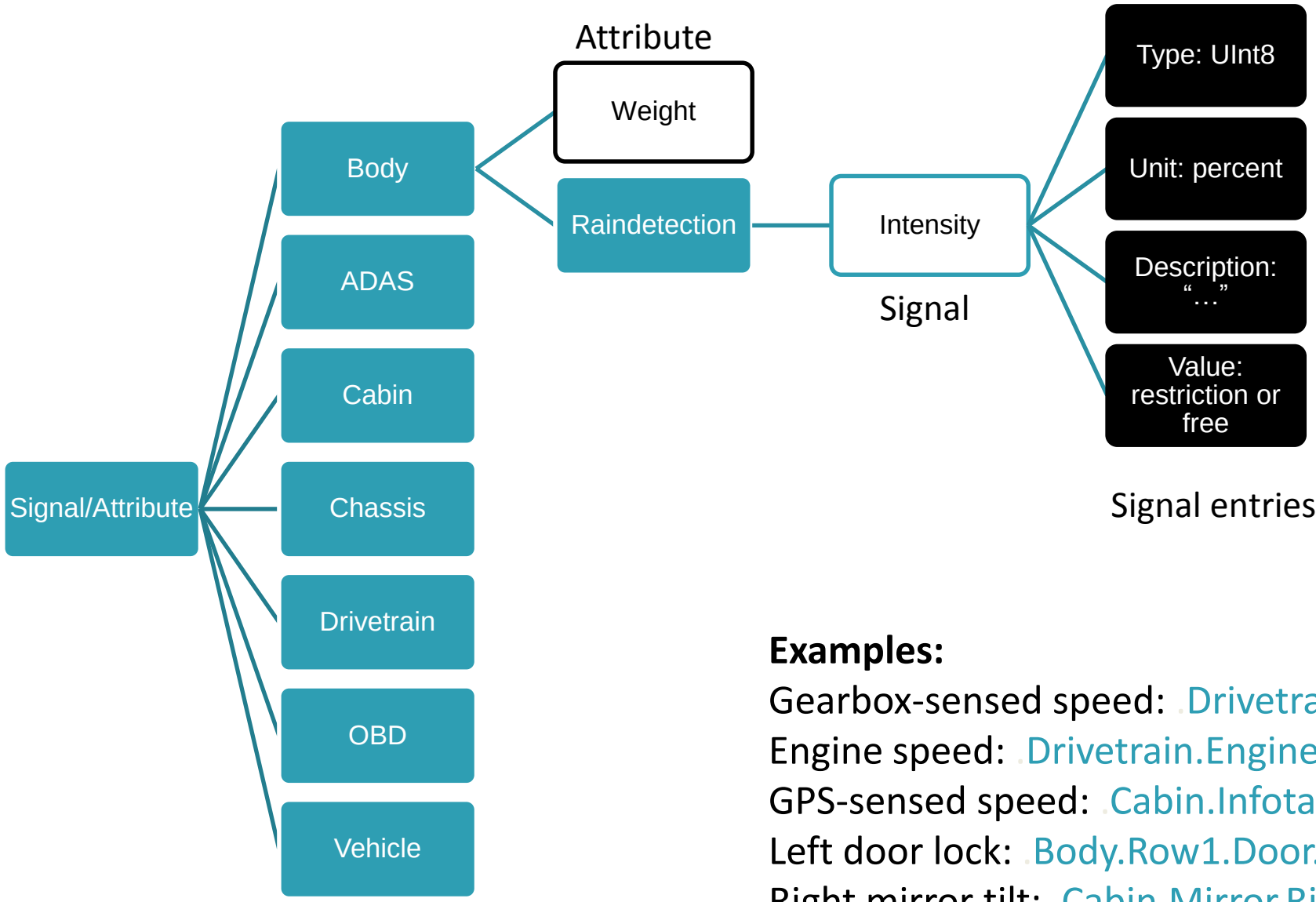


Figure:

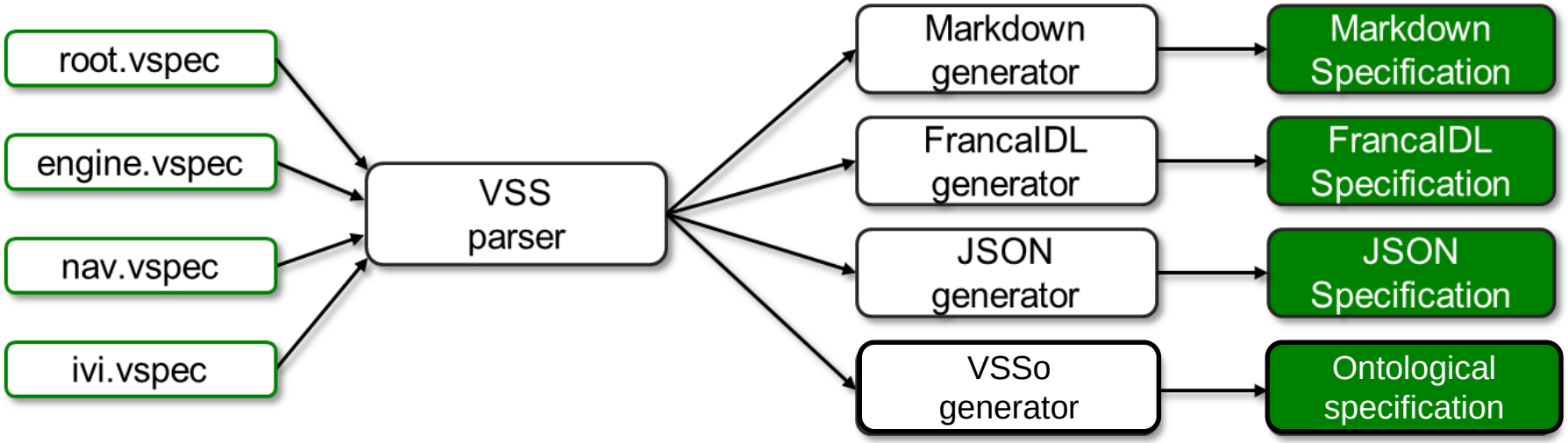
- 451 branches
- 1103 leaves:
  - 43 attributes
  - 1060 signals: including
    - (700 seat-related),
    - 268 with unit

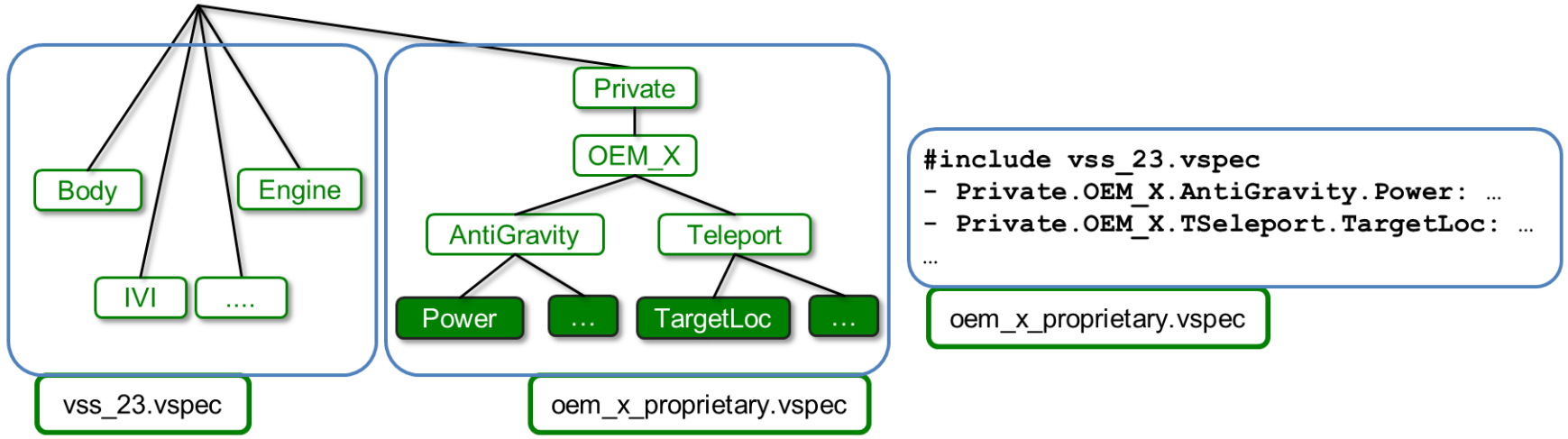
Examples:

Gearbox-sensed speed: Drivetrain.Transmission.Speed
Engine speed: Drivetrain.Engine.Speed
GPS-sensed speed: Cabin.Infotainment.Speed
Left door lock: Body.Row1.Door.Left.IsLocked
Right mirror tilt: Cabin.Mirror.Right.Tilt

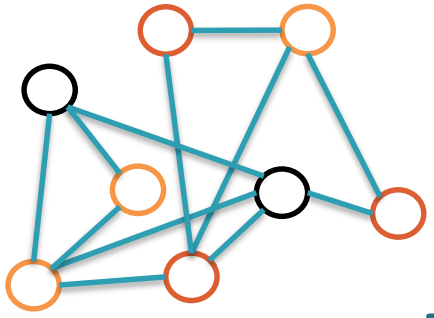
GENIVI®

Generation and extensions





VSSo: VSS ontology [3]

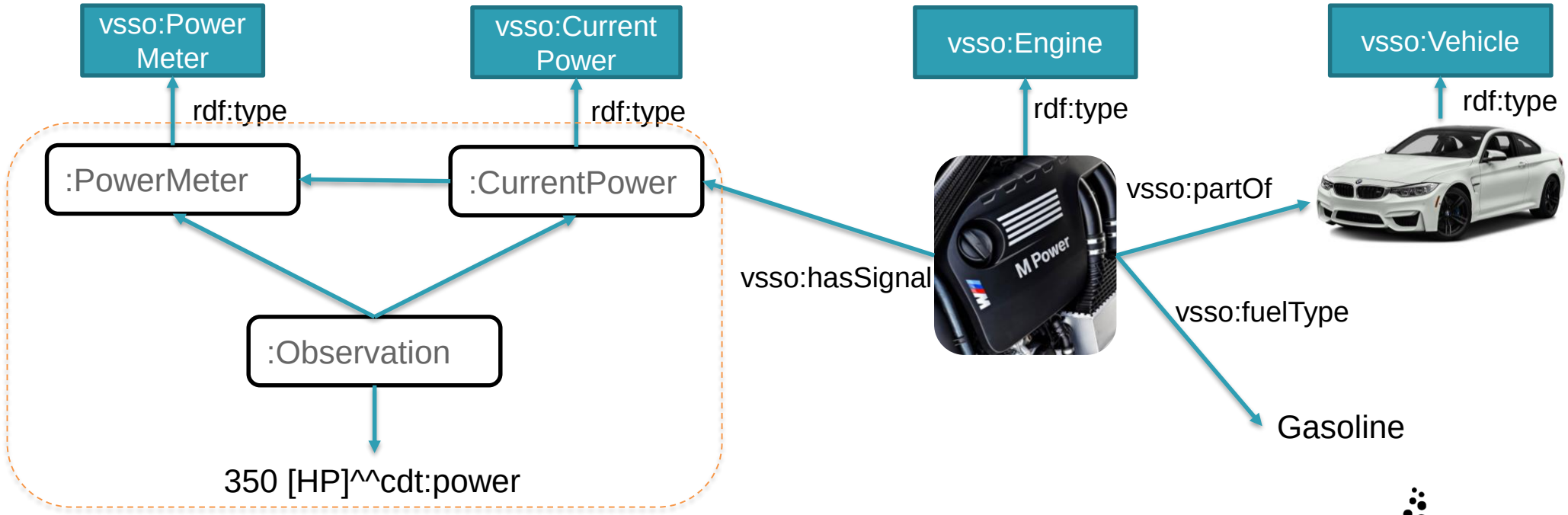


Graph representation of formal models of:

- Vehicles and their branches
- Sensors, actuators, signals and attributes

SOSA pattern [4]:

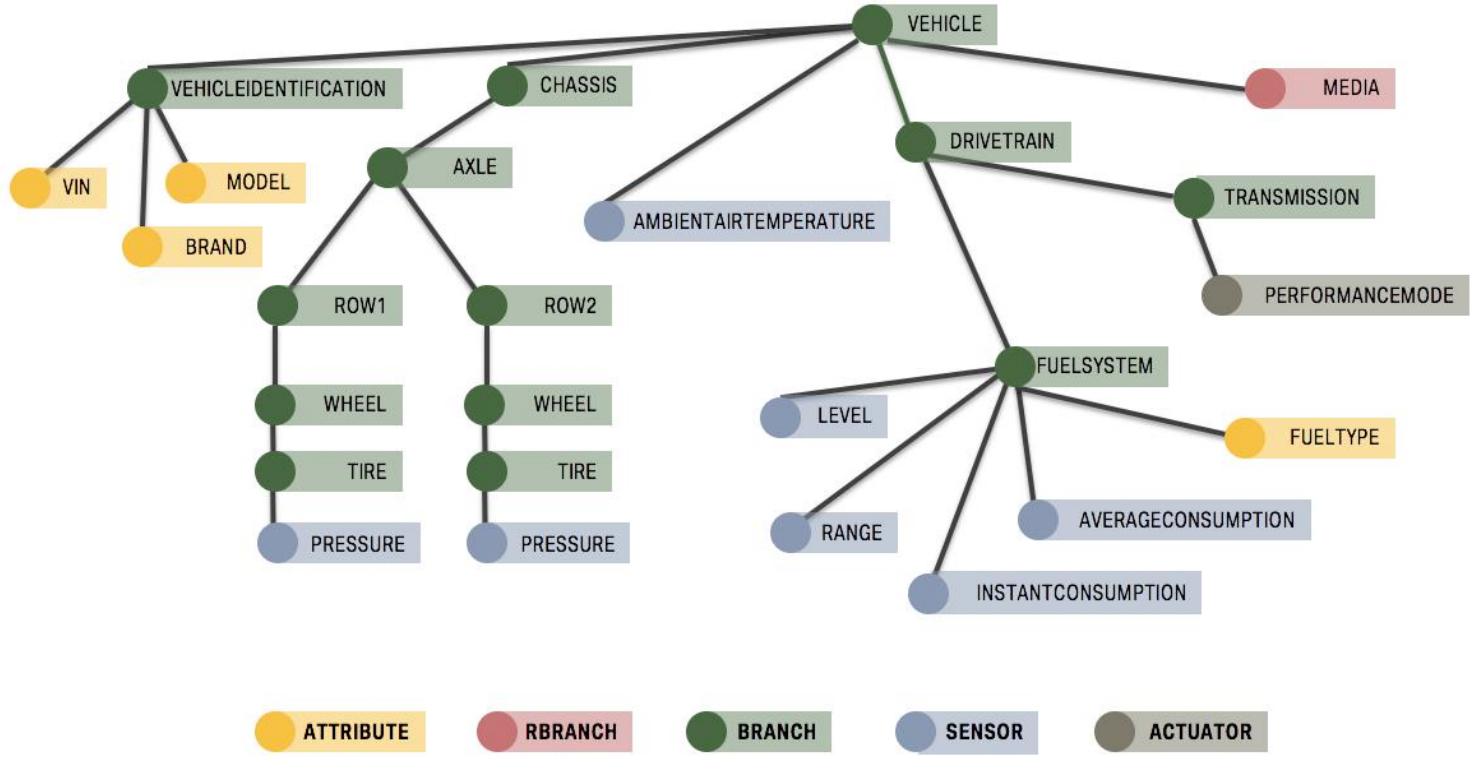
- Sensor,
- Observation,
- Sample,
- Actuator



[3] http://automotive.eurecom.fr/vsso
[4] https://www.w3.org/TR/vocab-ssn/

VSS 2.0

- A unified tree combining:
  - Static attributes
  - Sensors
  - Actuators
- Simpler position models:
  - Observe wheels in Row[1,2
  - Get window position in [LEFT,RIGHT]
- Rbranch:
  - Resource branch
  - Adapted for collections



Toward a Common Vehicle Data Model

Discussion

Which data models are missing?

How is access managed?

Are there some signals that should allow for more than one unit to be fair to all OEMs?

What parts of the VSS-based ecosystem is based on a standard database of named signal?
How large parts are proprietary extensions?

Which approach and policies for future-proofing?

How the choice of technical specification can affect the result (performance, feasibility of ...)?

Next session at 13:45

The Value of Vehicle Data to Enterprises