



VSSo: the Vehicle Signal and Attribute Ontology

Benjamin Klotz & Raphaël Troncy

✉ klotz@eurecom.fr



**BMW
GROUP**

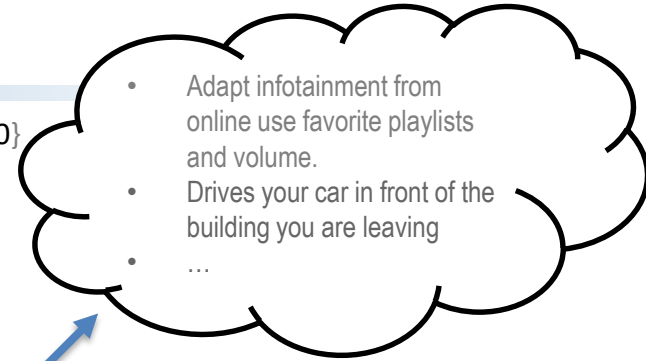
THE NEXT
100 YEARS



Context



Research `{"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}`



Temperature sensor

Adaptive cruise control

Front camera

Radar

Blind spot detection

Wheel speed sensor

Steering angle sensor

Oil temperature sensor

Vehicle height sensor

Park assistant

Tire pressure sensor

Requirements [1]

Get information about attributes and signals on connected vehicles:

Telematics/fleet management



What type of fuel does this car need?
What is the current gear?

Garage/diagnosis



What type of transmission does this car have?
How many different speedometers does this car contain?



E-commerce

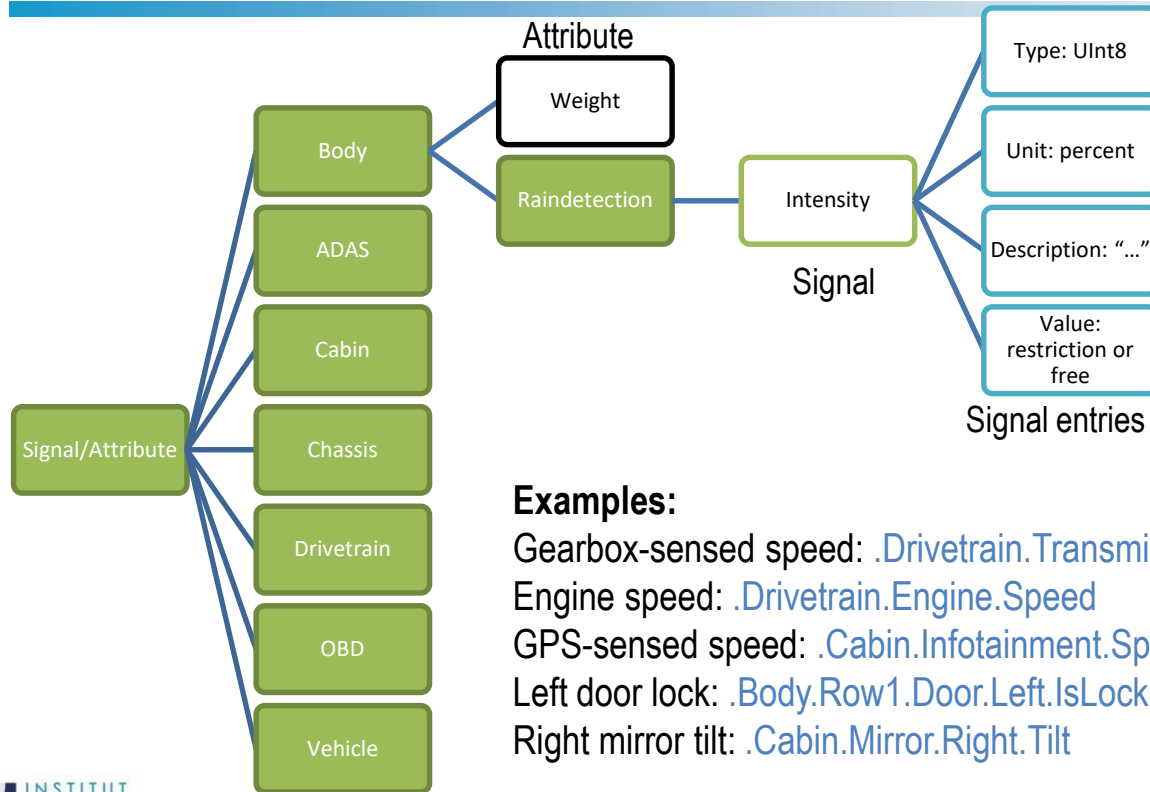
What is the model of this car?
How old is this car?



Seamless experience

What are the destination coordinates?
What is the local temperature on the driver side?

Vehicle signal specification (VSS) [2]



Figures (Apr 2018):

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

Examples:

Gearbox-sensed speed: `.Drivetrain.Transmission.Speed`

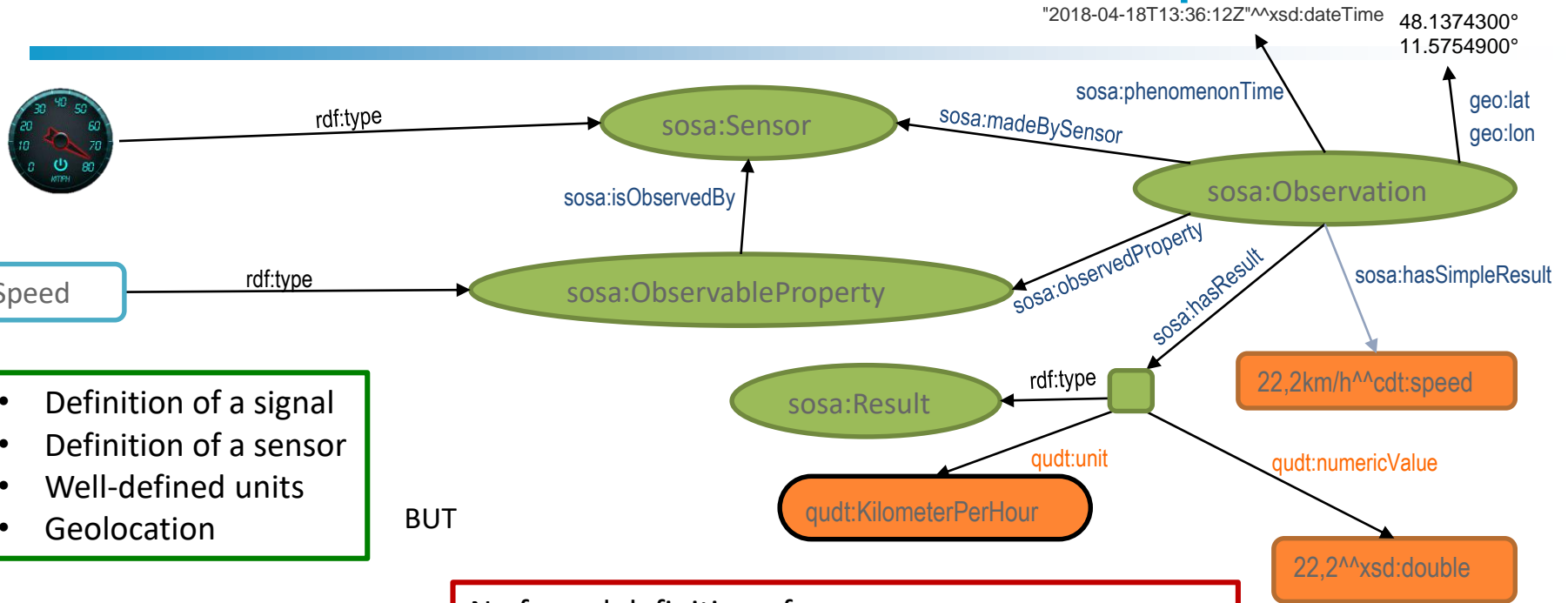
Engine speed: `.Drivetrain.Engine.Speed`

GPS-sensed speed: `.Cabin.Infotainment.Speed`

Left door lock: `.Body.Row1.Door.Left.IsLocked`

Right mirror tilt: `.Cabin.Mirror.Right.Tilt`

SOSA/SSN observation & sensor pattern

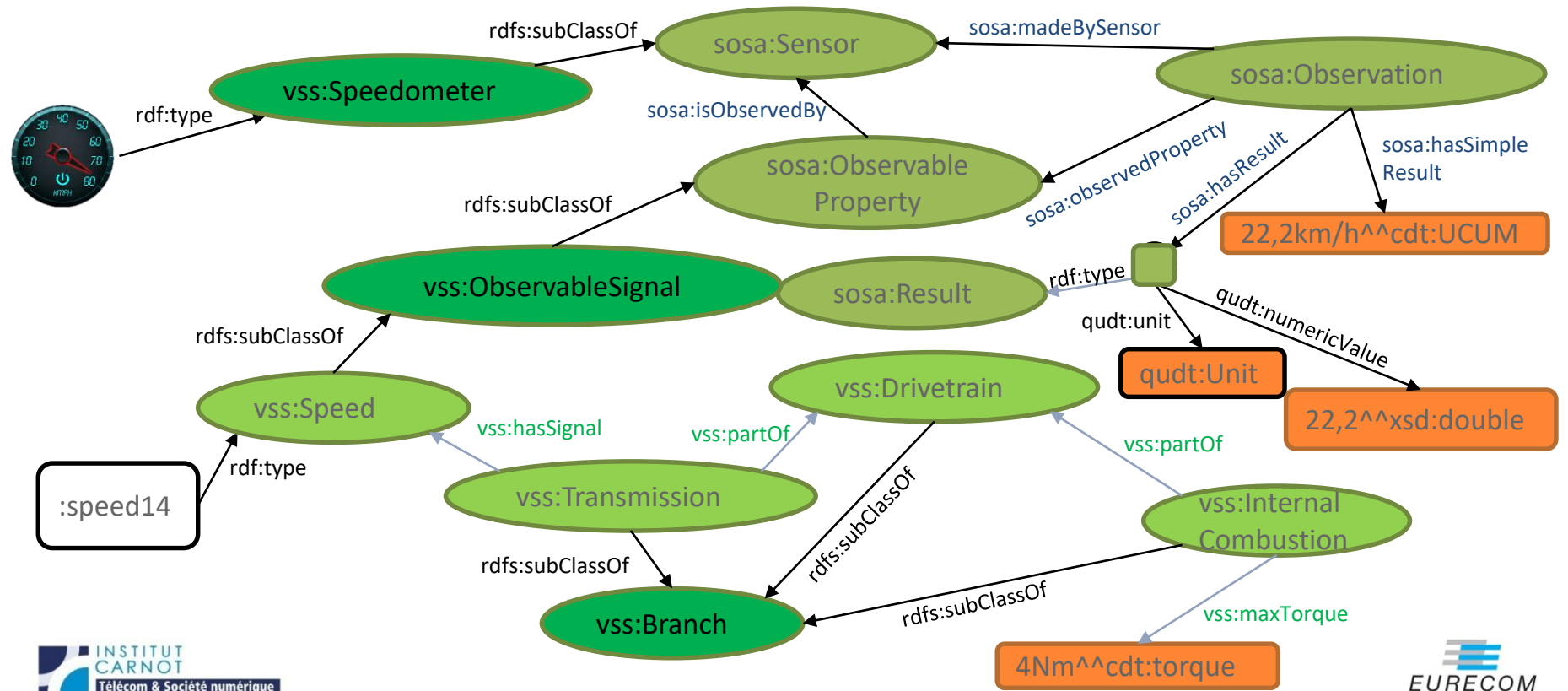


- Definition of a signal
- Definition of a sensor
- Well-defined units
- Geolocation

BUT

- No formal definition of:
 - “speed” or other observable properties
 - “speedometer” or other car sensors/actuators
 - “car” or vehicle parts

SSN/SOSA with a VSS ontology



VSSo: a Vehicle Signal Ontology

Map to existing Ontologies

- SSN/SOSA
- QUDT (unit)
- custom datatypes [4]

Generate definition
of VSS concepts

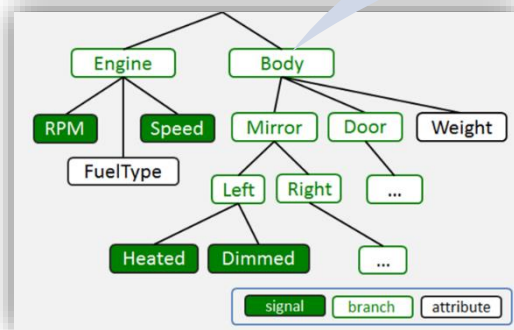
Add
sensors and
actuators

Fixing
problems

Manually validate and clean the
generated ontology

VSSontology [3]

VSS

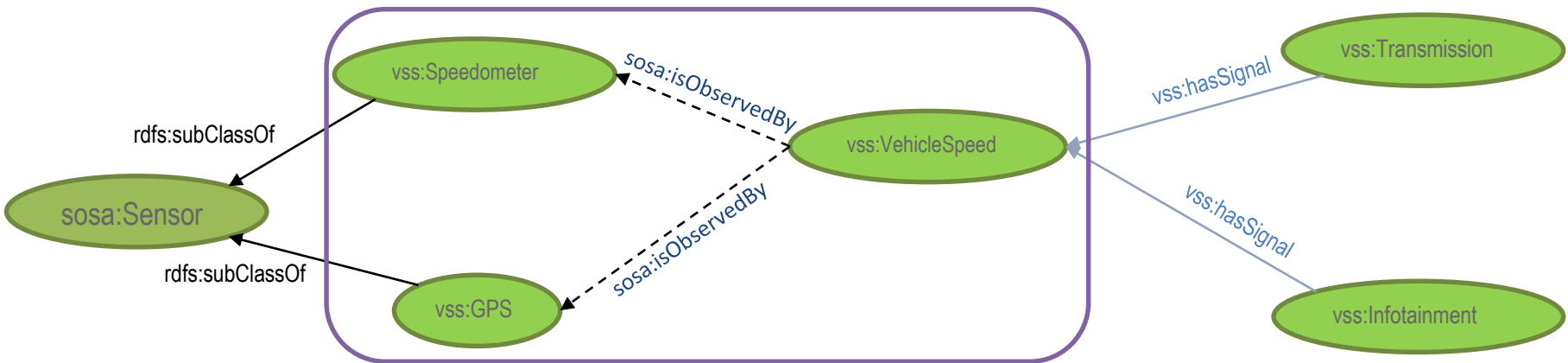


Modeling issues and resolutions

1. VSS concepts have now unique names
2. All signals are attached to (virtual) sensors or actuators
3. All position-dependent branches have a property “position”

#1: Uniqueness of names

- Some signals represent the same phenomenon, but sensed by different sensors
 - Ex: Drivetrain.Speed (sensed by the **gearbox**) and Infotainment.Speed (sensed by the **GPS**)



“vss:VehicleSpeed” is a unique phenomenon

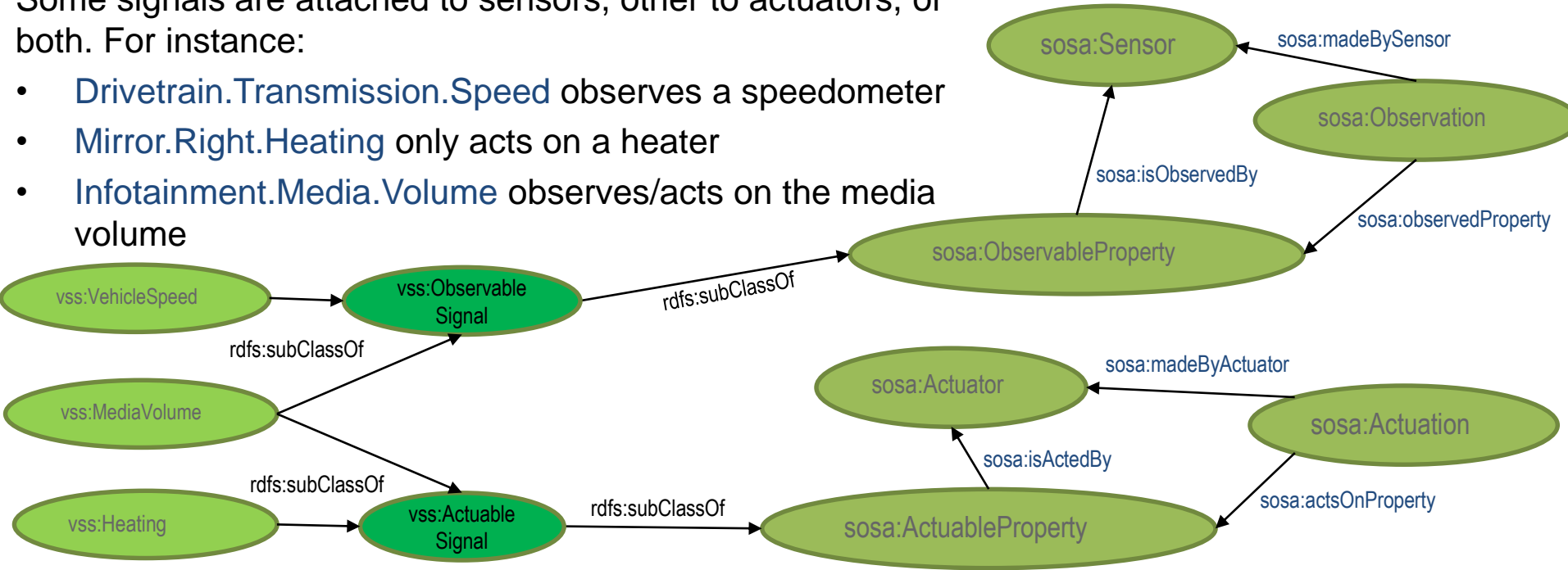
- observed by different sensors
- Producing different signals

Names are clarified to avoid homonymy

#2: Signals are attached to (virtual) sensors or actuators

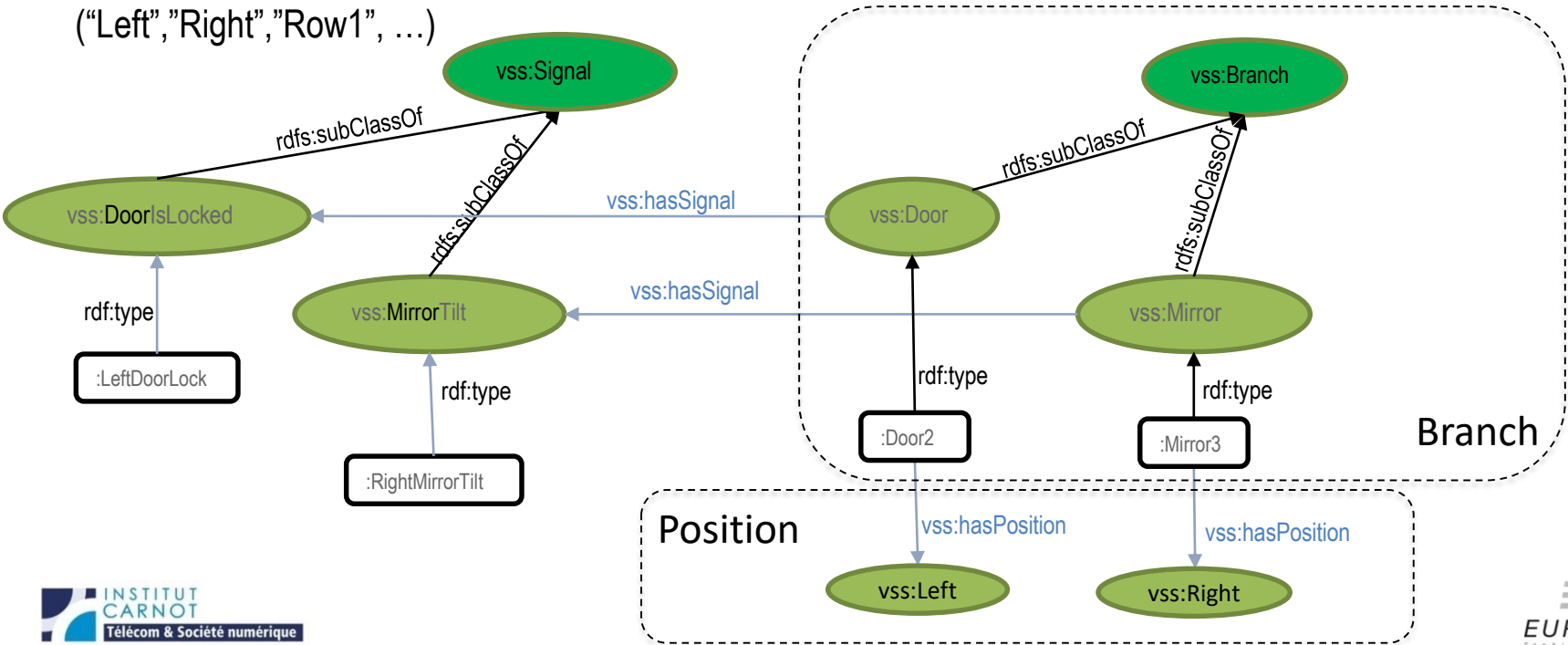
Some signals are attached to sensors, other to actuators, or both. For instance:

- **Drivetrain.Transmission.Speed** observes a speedometer
- **Mirror.Right.Heating** only acts on a heater
- **Infotainment.Media.Volume** observes/acts on the media volume

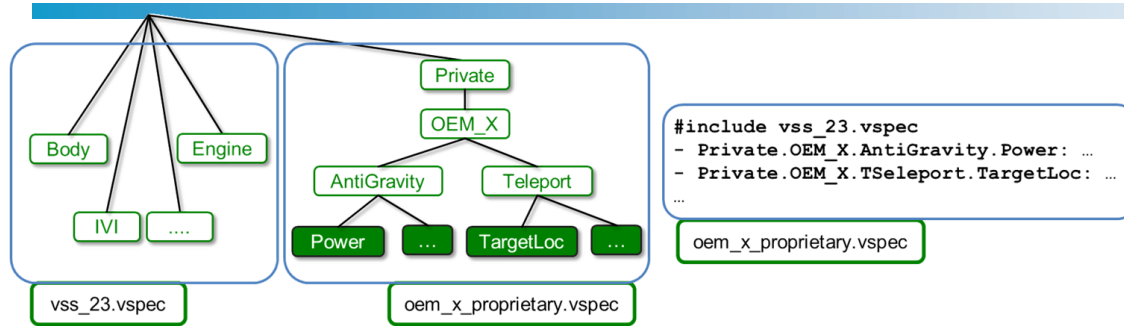


#3: position ≠ branch

- Ex: Door.Left.IsLocked, Mirror.Right.Tilt
- Branches vss:Door and vss:Mirror have **vss:hasPosition** property with limited potential values (“Left”, “Right”, “Row1”, ...)



VSS private branch



Private branch

- OEM-specific concepts
- Extension of VSS
- Merged into VSS when generating specifications

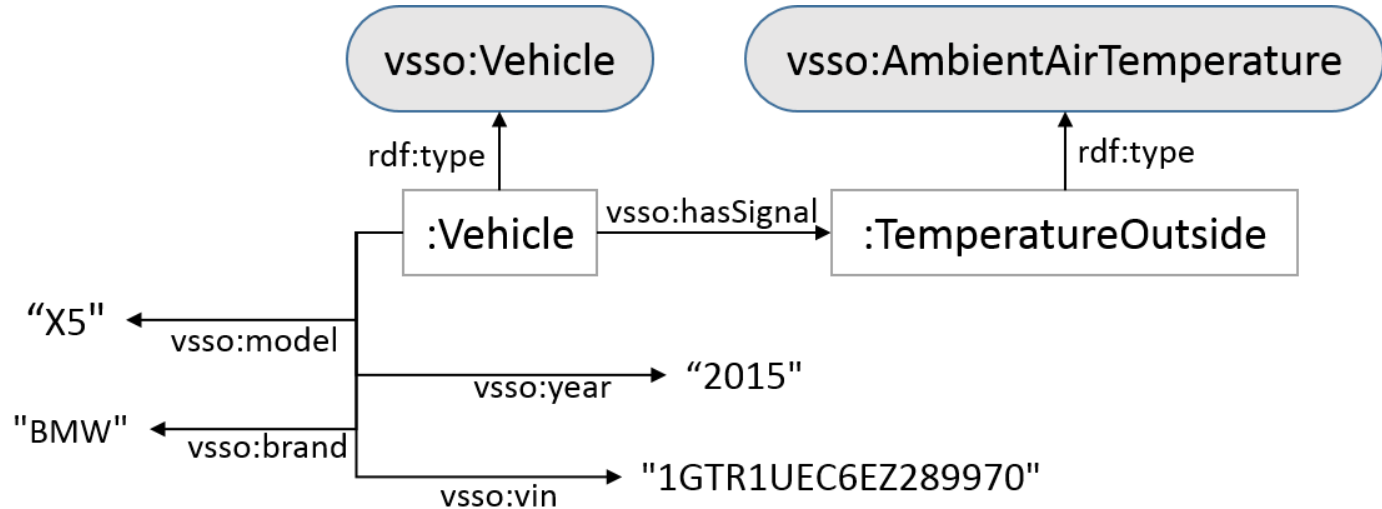
Private OEM-specific ontology cookbook:

1. Write VSS-compliant specification of private concepts (new signals, attributes and branches)
 - Follow the VSS policies just as when creating a private branch
2. Generate the ontology using the existing tool
3. Validate the ontology
 - Check the unicity of concepts and definitions (in the private branch and if possible with VSSo)
4. Define a private namespace for your ontology integrating VSSo

Evaluation

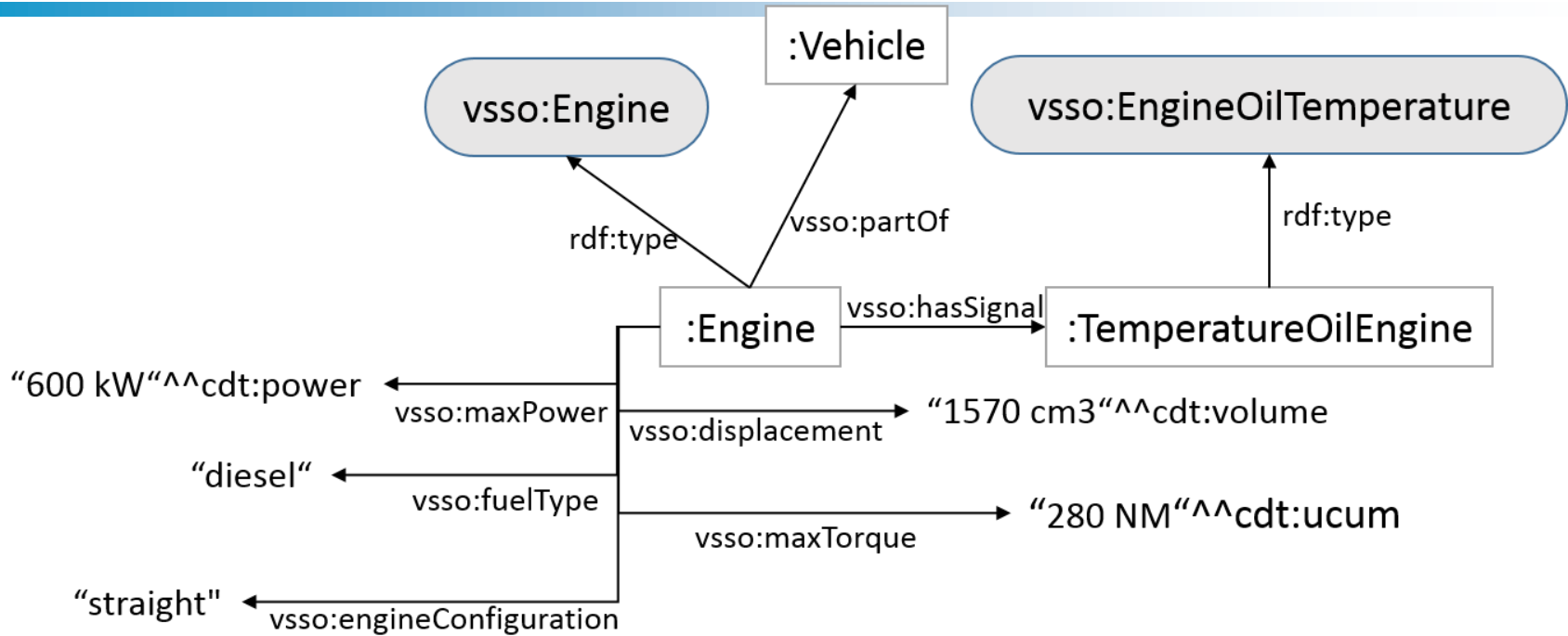
- VSSo expressivity: requirements can be expressed in terms of SPARQL queries
 - [What are the dimension of this car?](#)
 - [What is the current temperature on the driver side?](#)
 - Endpoints: <http://automotive.eurecom.fr/simulator/query>
<http://automotive.eurecom.fr/simulator/fleetquery/>
- VSSo extension mechanism is currently under test with real use cases at BMW
 - Pol definitions in the GPS and distance to the destination
 - Streaming of sensors

Examples (1)



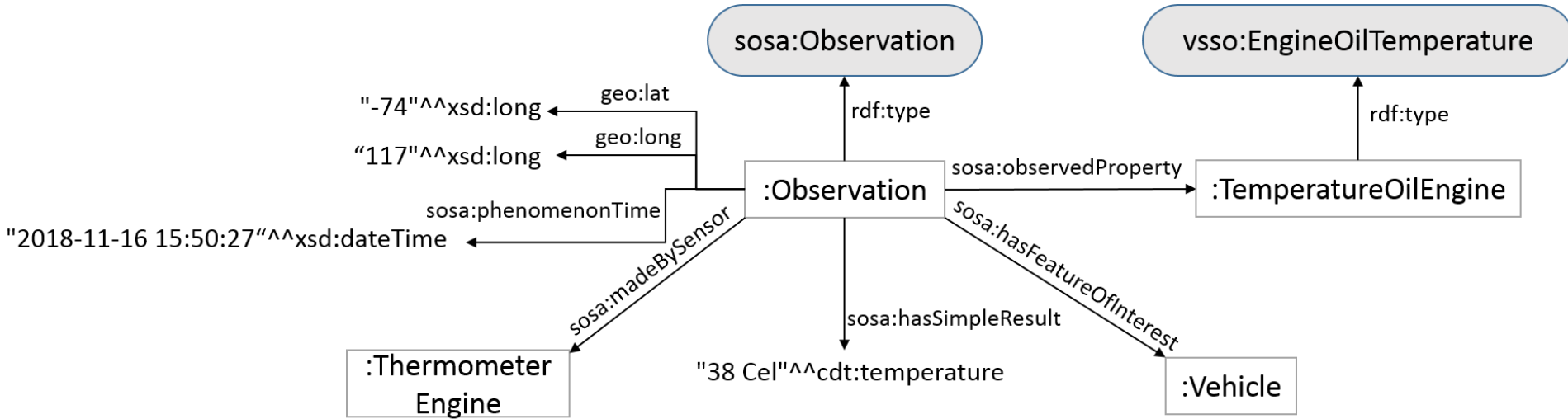
A vehicle with attributes and a single signal

Examples (2)



A branch (:Engine) with attributes and a single signal

Examples (3)



A single observation of a signal

Conclusion

VSSo: a Vehicle Signal and Attribute ontology

- 483 classes (~300 signals); 63 properties (~50 attributes)
- Documentation: <http://automotive.eurecom.fr/vsso> (v1.21)
- Recommended prefix: vsso
- Re-use SSN/SOSA modeling patterns



What should be materialized in graph data?
Continuous time series of sensor data?

Future work @W3C

- Move all documentation at <https://github.com/w3c/vsso/>
- Sync VSSo with the latest GENIVI VSS
 - Reference: https://github.com/GENIVI/vehicle_signal_specification/
 - Make use of VSS layers
- Share large datasets using the VSSo model for fast prototyping:
 - Use case: modeling, expressivity, search
 - Use case: graph data mining, link prediction
- Create tooling for connected cars