

VSS ONTOLOGY EXTENSION PROPOSALS

Dr. Felix Lösch

Chief Digital Office for Mobility Solutions
Robert Bosch GmbH

VSS Ontology

Change & Extension Proposals

▶ CP1: Treatment of Vehicle Component and Vehicle Signals

- ▶ Treat Vehicle Component and Vehicle Signals as instances of classes VehicleComponent and VehicleProperty instead of classes

▶ CP2: Treatment of Vehicle class in VSSo

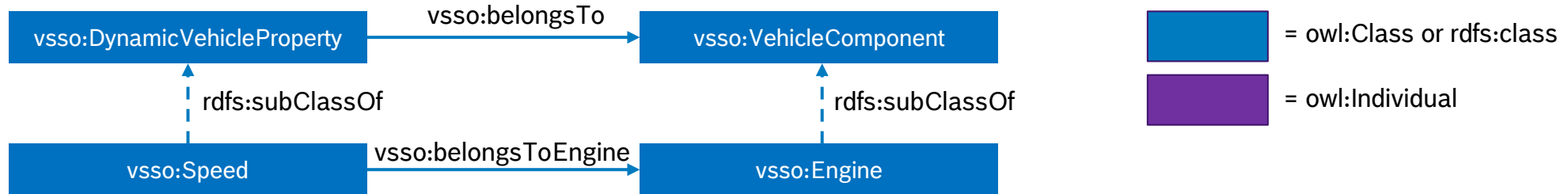
- ▶ Vehicle class represents both VehicleComponent Root and Vehicle Instance
- ▶ We propose to differentiate these two
- ▶ → Create class “VehicleInstance” which describes concrete vehicles
- ▶ → VehicleComponent root will become an instance of the “VehicleComponent” class

▶ EP1: Introduction of classes for Property Types

- ▶ VSS 2.2 adds enumeration types → we need a class for handling enumeration, i.e., EnumerationProperty
- ▶ Not every property has a unit in VSS → we should add a class called NumericProperty

VSSo – Subclassing vs. Instances

CP1: VSSo – Current Proposal (Subclasses)



vss:belongsTo
rdfs:domain vss:DynamicVehicleProperty
rdfs:range vss:VehicleComponent

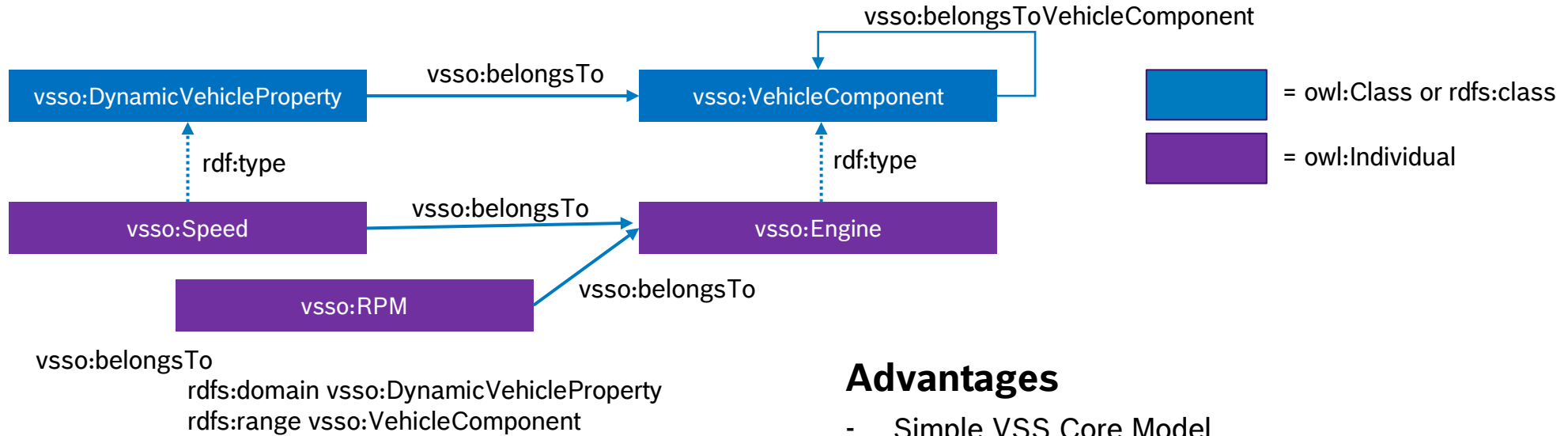
vss:belongsToEngine
rdfs:domain vss:Speed
rdfs:range vss:Engine
rdfs:subPropertyOf vss:belongsTo

Disadvantages

- Many subclasses required
- Subclasses do not introduce new data properties
- Individual object properties required for linking vehicle property subclasses to Vehicle component subclasses

VSSo – Subclassing vs. Instances

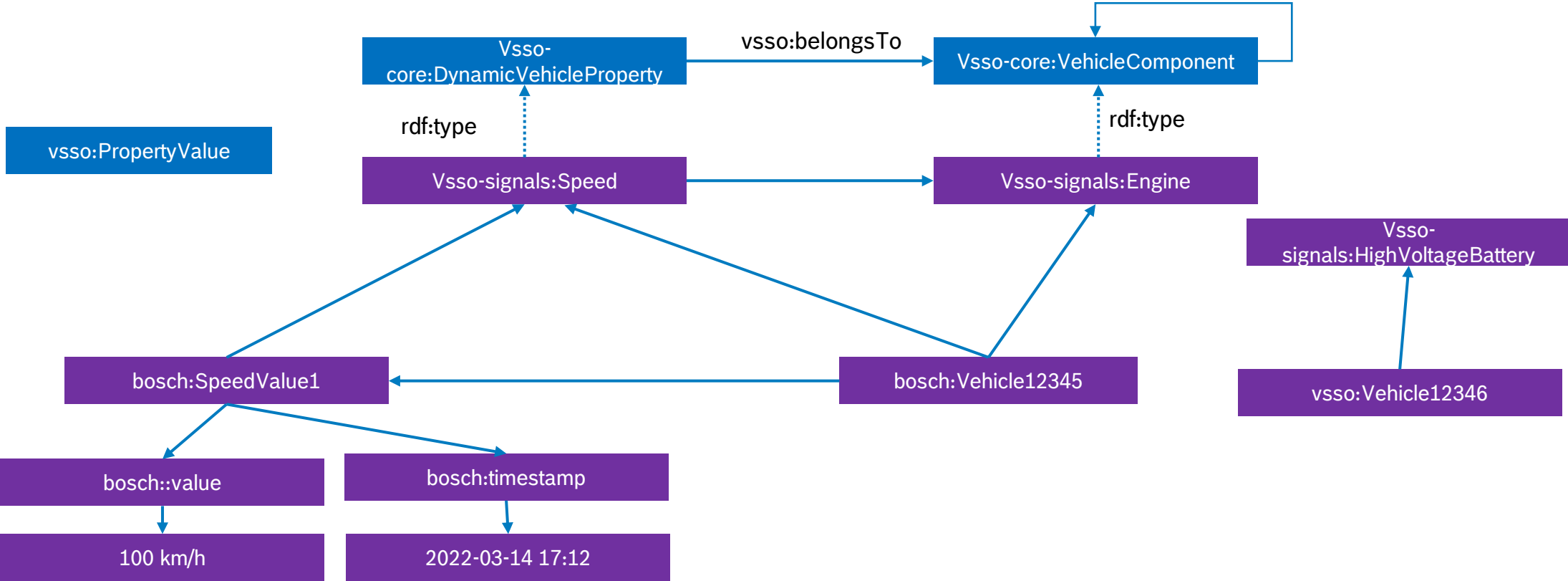
CP1: VSSo – New Proposal (Instances)



Advantages

- Simple VSS Core Model
- Reuse of `belongsTo` to link vehicle property instances to `VehicleComponent` instances
- Much easier handling when VSS evolves

VSSo – Subclassing vs. Instances



VSSo – Subclassing vs. Instances

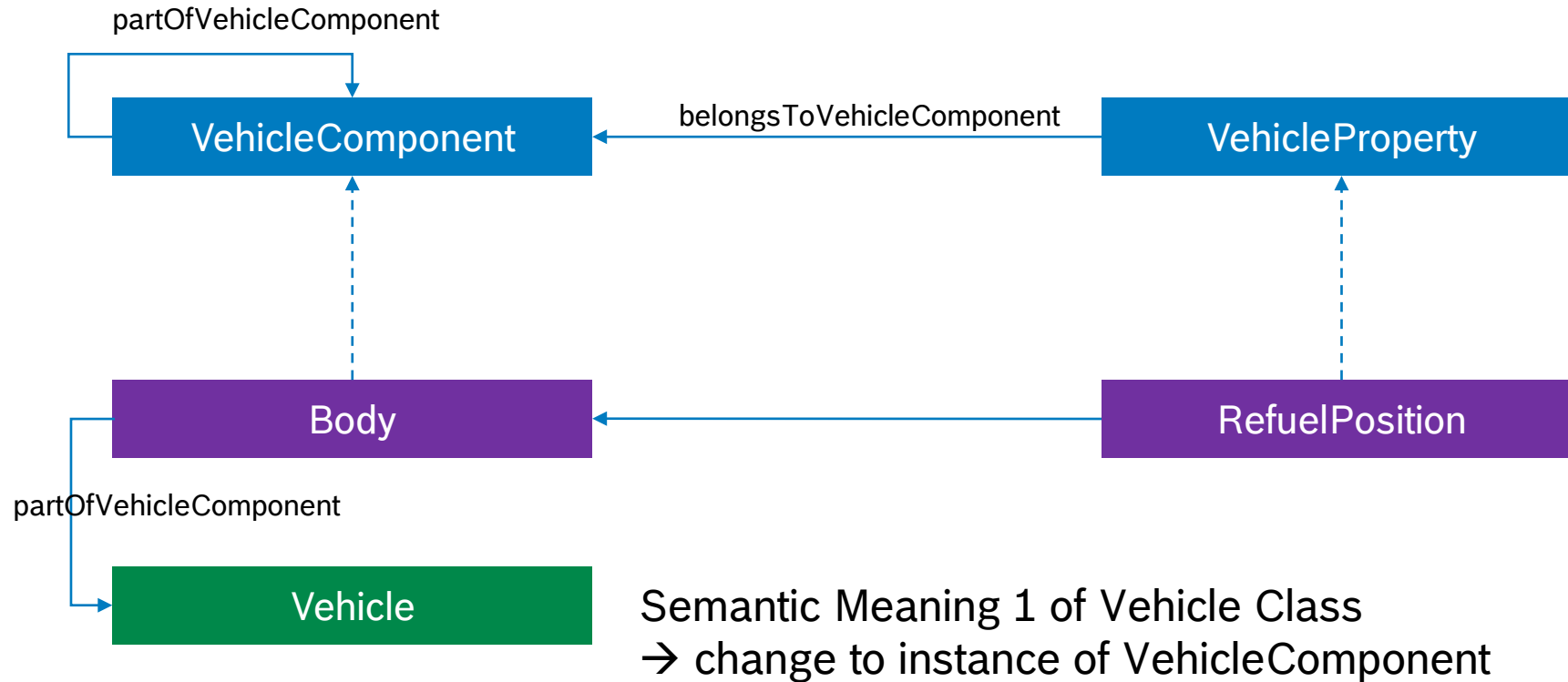
Discussion on Subclassing vs. instances (VSSo WS 14.03.22)

- ▶ DynamicVehicleProperty: Proposal to realize as instances
- ▶ StaticVehicleProperty: Proposal to realize as subclasses to instantiate for concrete values
- ▶ VehicleComponent: Proposal to realize as instances
- ▶ Vspec2ttl (VSS Tools): define rules /exceptions how to generate VSS signals in VSSo

- ▶ Open points:
 - ▶ VSS Arrays
 - ▶ VSS Instantiations
 - ▶ Units
 - ▶ Property Type (quantity / enumeration)

VSS Ontology

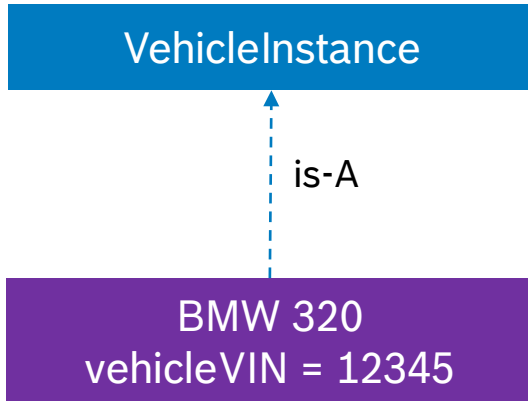
CP2: Treatment of Vehicle Class in VSSo



VSS Ontology

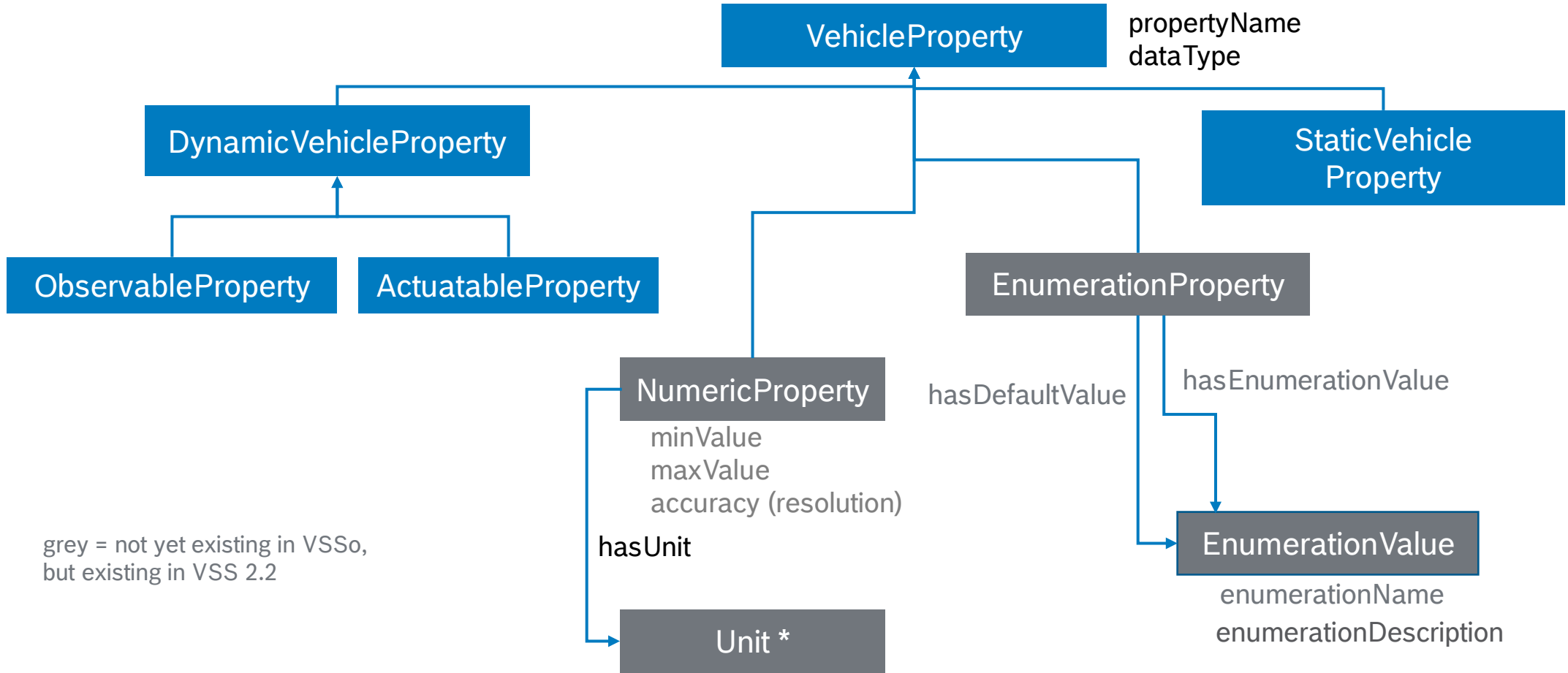
CP2: Treatment of Vehicle Class in VSSo

VehicleInstances shall be described as instances of class “VehicleInstance”



VSS Ontology

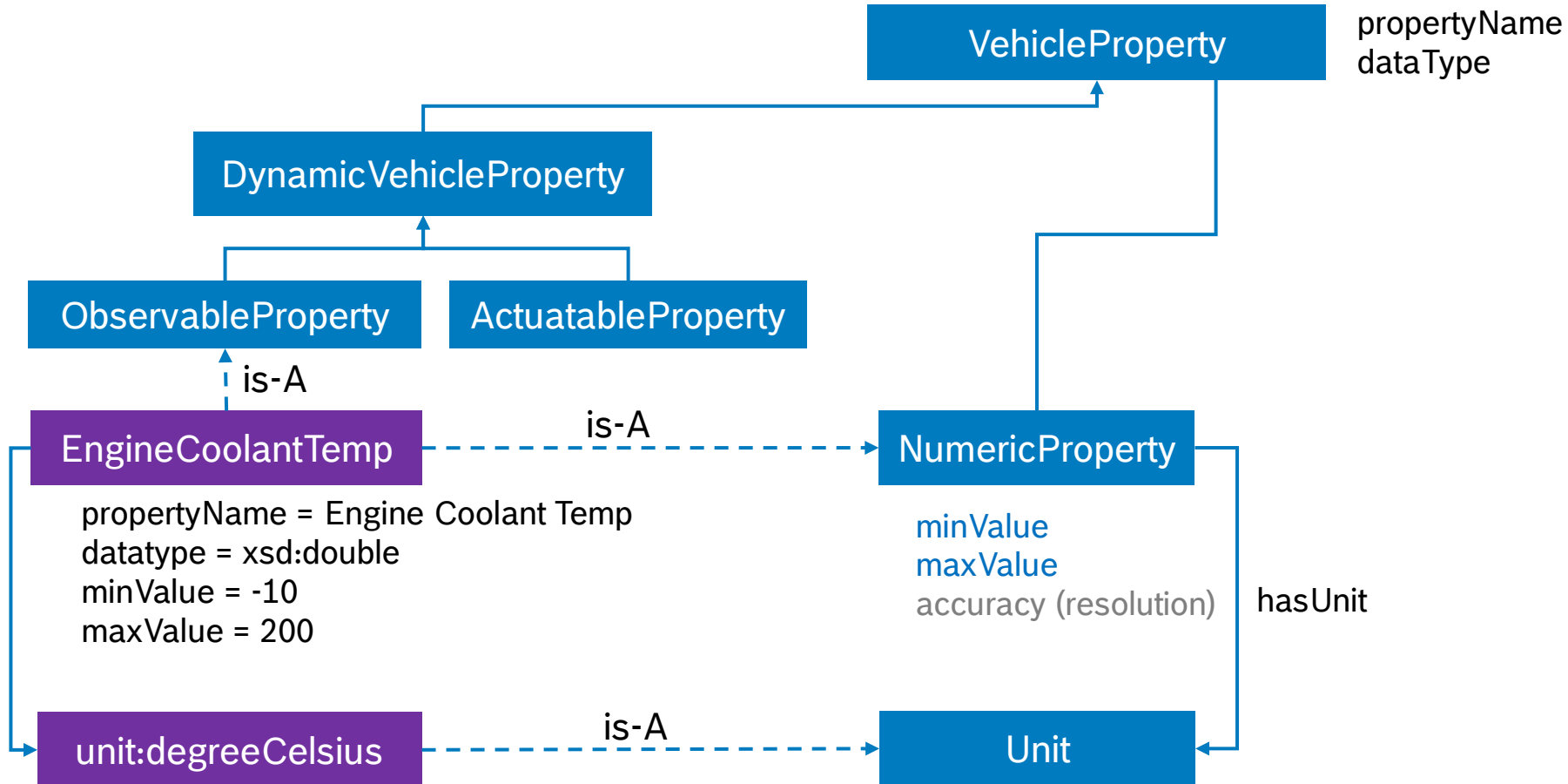
EP1: Additional Property Types



* We should agree on unit ontology (e.g. QU-REC20)

VSS Ontology

EP1: Example for NumericProperty



VSS Ontology

EP1: Example for Enumeration Property

