

D Annex D: Sample Problem

(Informative)

D.1 Purpose

The purpose of this annex is to illustrate how the nominal usage of SysML model can support the specification, analysis, and design of a system. This focuses on using the basic features of the language in building a system model, and then rendering the model as SysML diagrams.

While this annex focuses on diagrams, the reader should remember that these diagrams are rendered from an integrated model using tooling available at the time this specification was published. No post-processing has been applied to these diagrams; they appear just as they are represented in the modeling tool. This was done to streamline the generation and maintenance of this appendix. Effort has been made to suppress tool-specific graphic artifacts on these diagrams, but some such artifacts may be unavoidable.

D.2 Scope

The scope of this example is a single SysML system model that can be used to render at least one diagram of each diagram type. The intent is to model simplified fragments of the problem to illustrate how a model can be constructed, and to demonstrate some of the possible inter-relationships among the model elements. The sample problem is not intended to exercise all of the features of the language. The reader should refer to the individual clauses for more detailed features of the language. The diagrams rendered for representing a particular aspect of the model, and the ordering of the diagrams are intended to be representative of applying a nominal systems engineering process, but is not intended to endorse any specific process or methodology.

D.3 Problem Summary

The sample problem describes the use of SysML as it applies to the development of an automobile, in particular a Hybrid gas/electric powered Sport Utility Vehicle (SUV). This problem is interesting in that it has inherently conflicting requirements, viz. desire for fuel efficiency, but also desire for large cargo carrying capacity and off-road capability. Technical accuracy and the feasibility of the actual solution proposed were not high priorities. This sample problem focuses on design decisions surrounding the power subsystem of the hybrid SUV; the requirements, performance analyses, structure, and behavior.

This annex is structured to show each diagram in the context of how it might be used on such an example problem. The first sub clause shows SysML diagrams as they might be used to establish the system context; establishing system boundaries, and top level use cases. The next sub clause is provided to show how SysML diagrams can be used to analyze top level system behavior, using sequence diagrams and state machine diagrams. The following sub clause focuses on use of SysML diagrams for capturing and deriving requirements, using diagrams and tables. A sub clause is provided to illustrate how SysML is used to depict system structure, including block hierarchy and part relationships. The relationship of various system parameters, performance constraints, analyses, and timing diagrams are illustrated in the next sub clause. A sub clause is then dedicated to illustrating definition and depiction of interfaces and flows in a structural context. The final sub clause focuses on detailed behavior modeling, functional and flow allocation.

D.4 Diagrams

D.4.1 Package Overview (Structure of the Sample Model)

D.4.1.1 Package Diagram - Applying the SysML Profile

As shown in [Fig. D.1](#) the HSUVModel is a package that represents the user model. The SysML Profile is applied to this package in order to include stereotypes from the profile. The HSUVModel also requires the use model libraries, such as the SI Units Types model library. The model libraries are imported into the user model as indicated.

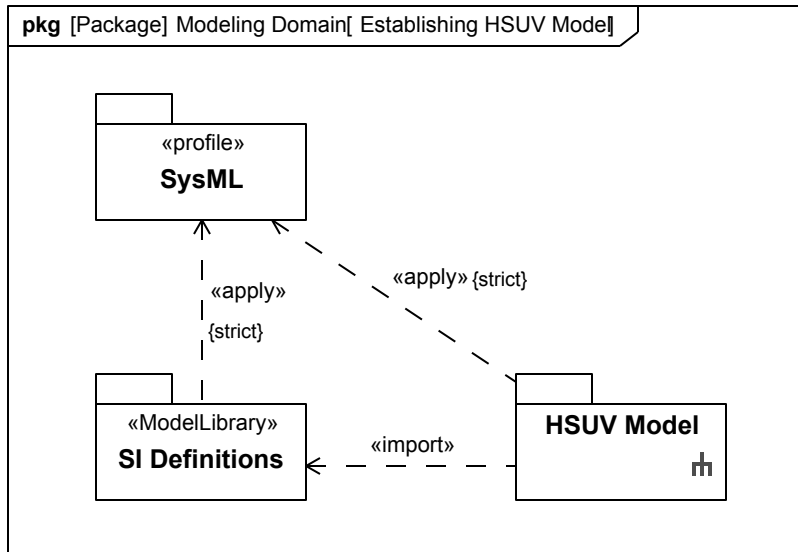


Figure D.1. Establishing the User Model by importing and applying SysML Profile & Model Library (Package Diagram)

Fig. D.2 shows the specification of automotive units and valueTypes employed in this sample problem.

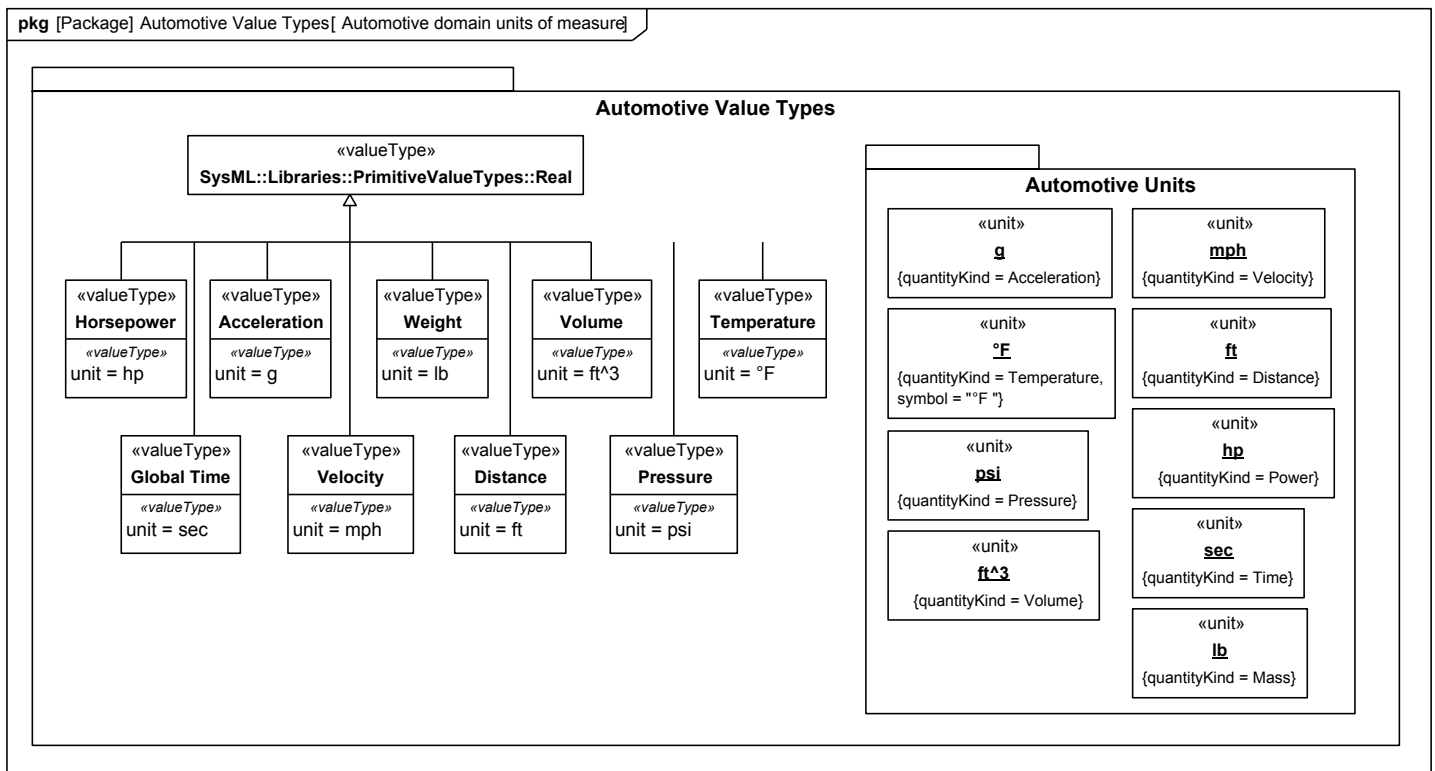


Figure D.2. Defining value Types and units to be used in the Sample Problem

D.4.1.2 Package Diagram - Showing Package Structure of the Model

The package diagram Fig. D.3 shows the structure of the model for this sample problem. Model elements are contained in packages, and relationships between packages (or specific model elements) are shown on this diagram. The relationship between the views (OperationalView and PerformanceView) and the rest of the user model are explicitly expressed using the «import» relationship. Note that the «view» models contain no model elements of their own, and that changes to the model in other packages are automatically updated in the Operational and Performance Views.

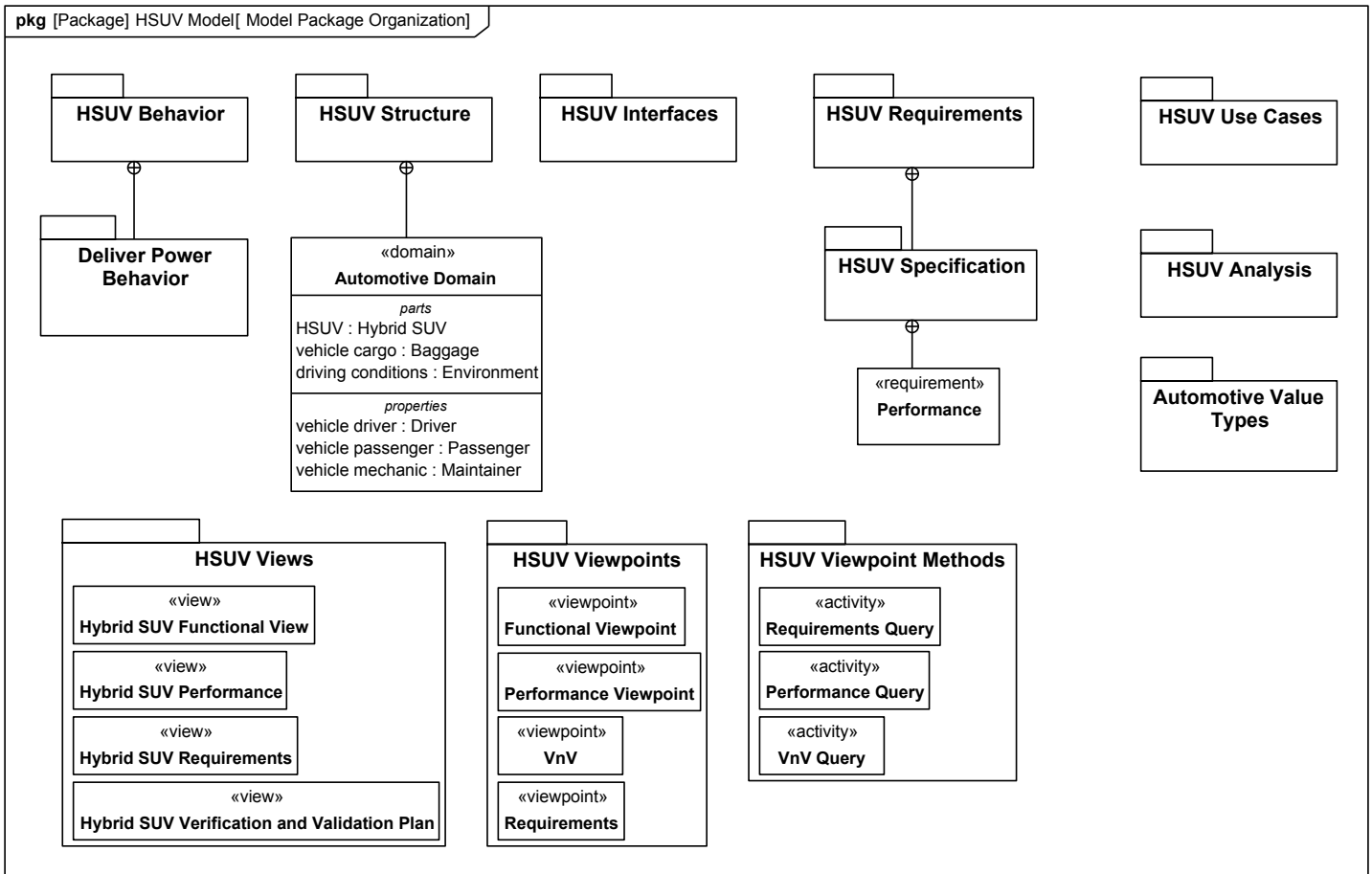


Figure D.3. Model Package Organization

D.4.2 Setting the Context (Boundaries and Use Cases)

D.4.2.1 Internal Block Diagram - Setting Context

The term “context diagram,” in [Fig. D.4](#), refers to a user-defined usage of an internal block diagram, which depicts some of the top-level entities in the overall enterprise and their relationships. The diagram usage enables the modeler or methodologist to specify a unique usage of a SysML diagram type using the extension mechanism described in Annex A, “Diagrams.” The entities are conceptual in nature during the initial phase of development, but will be refined as part of the development process. The «system» and «external» stereotypes are user defined, not specified in SysML, but help the modeler to identify the system of interest relative to its environment. Each model element depicted may include a graphical icon to help convey its intended meaning. The spatial relationship of the entities on the diagram sometimes conveys understanding as well, although this is not specifically captured in the semantics. Also, a background such as a map can be included to provide additional context. The associations among the classes may represent abstract conceptual relationships among the entities, which would be refined in subsequent diagrams. Note how the relationships in this diagram are also reflected in the Automotive Domain Model Block Definition Diagram, [Fig. D.15](#), which is rendered from the same underlying model.

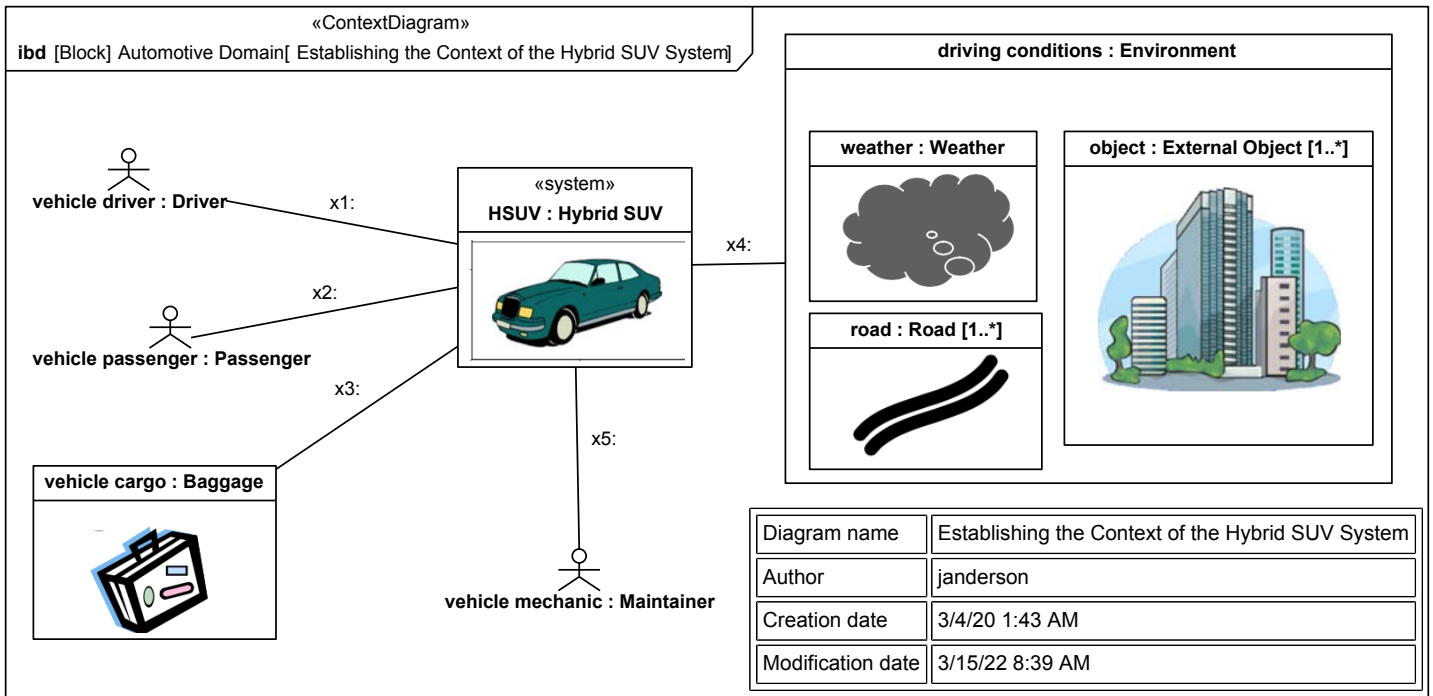


Figure D.4. Establishing the Context of the Hybrid SUV System

D.4.2.2 Use Case Diagram - Top Level Use Cases

The use case diagram “Establishing Top Level Uses Cases” in Fig. D.5 depicts usage in the Automotive Domain. The subject (HybridSUV) and the actors (Driver, Registered Owner, Maintainer, Insurance Company, DMV) interact to realize the use case.

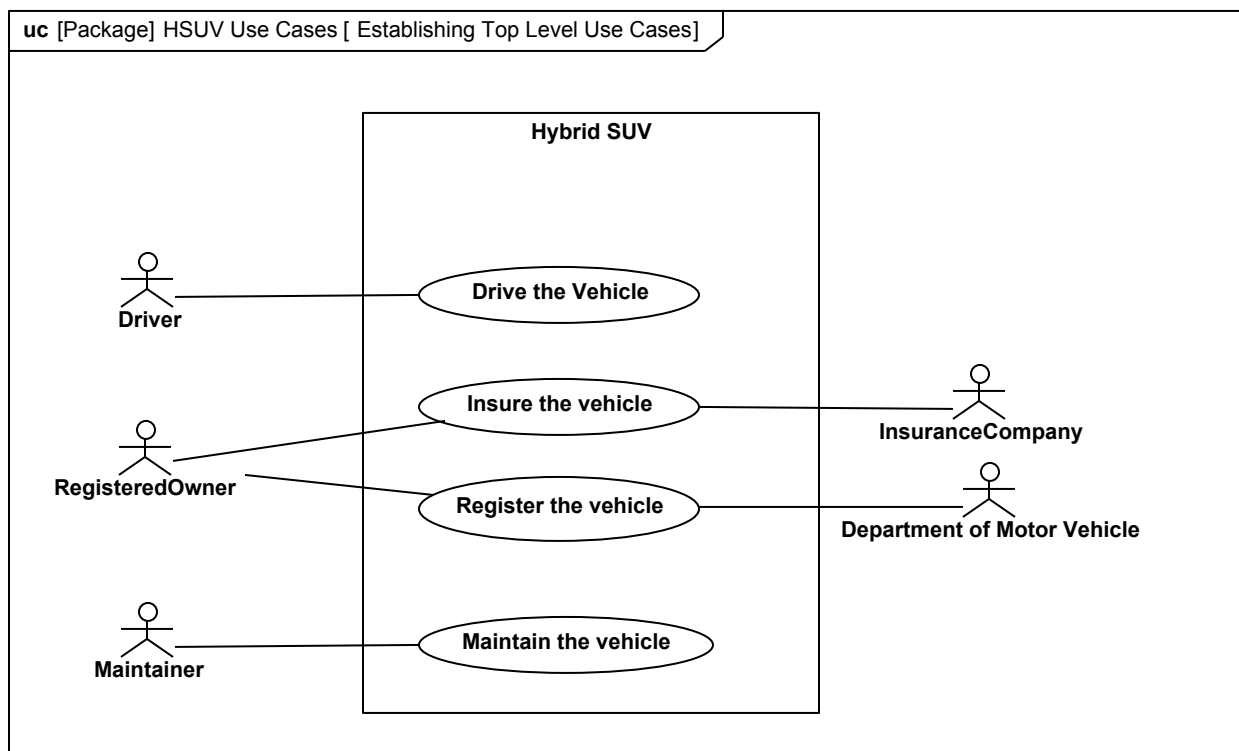


Figure D.5. Establishing Top Level Use Cases

D.4.2.3 Use Case Diagram - Optional Use Cases

Goal-level Use Cases associated with “Operate the Vehicle” are depicted in the following diagram. These use cases help flesh out the specific kind of goals associated with driving and parking the vehicle. Maintenance, registration, and insurance of the vehicle would be covered under a separate set of goal-oriented use cases.

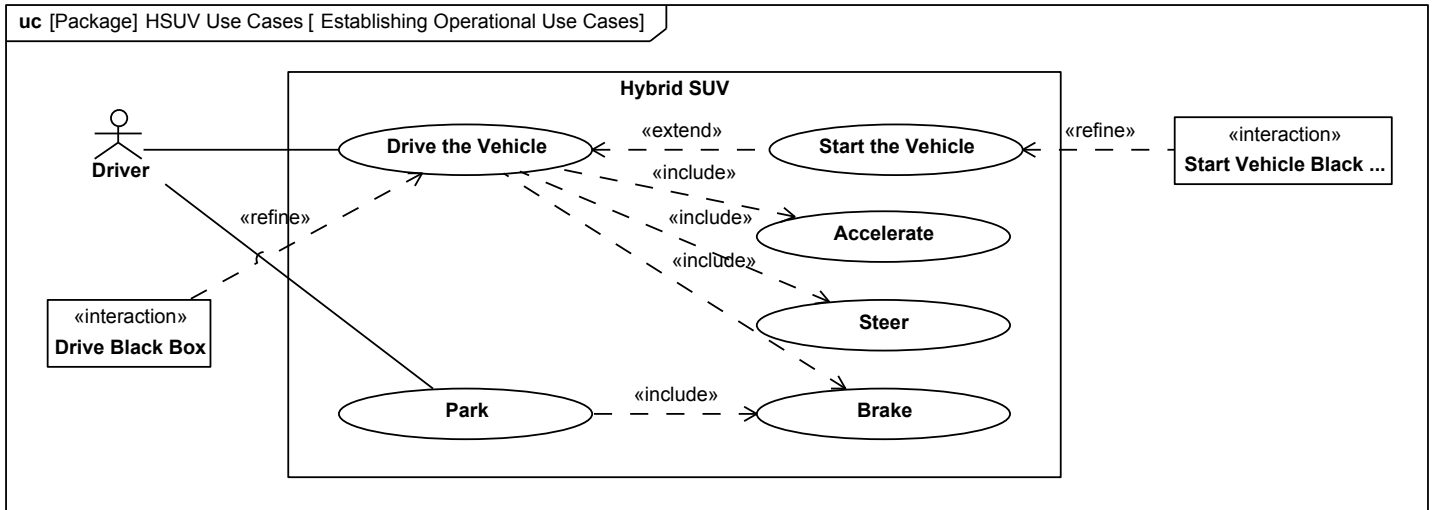


Figure D.6. Establishing Operational Use Cases

D.4.3 Elaborating Behavior (Sequence and State Machine Diagrams)

D.4.3.1 Sequence Diagram - Drive Black Box

[Fig. D.7](#) shows the interactions between driver and vehicle that are necessary for the “Drive the Vehicle” Use Case. This diagram represents the “DriveBlackBox” interaction, with is owned by the AutomotiveDomain block. “BlackBox” for the purpose of this example, refers to how the subject system (HybridSUV) interacts only with outside elements, without revealing any interior detail.

The conditions for each alternative in the alt controlSpeed sub clause are expressed in OCL, and relate to the states of the HybridSUV block, as shown in [Fig. D.8](#).

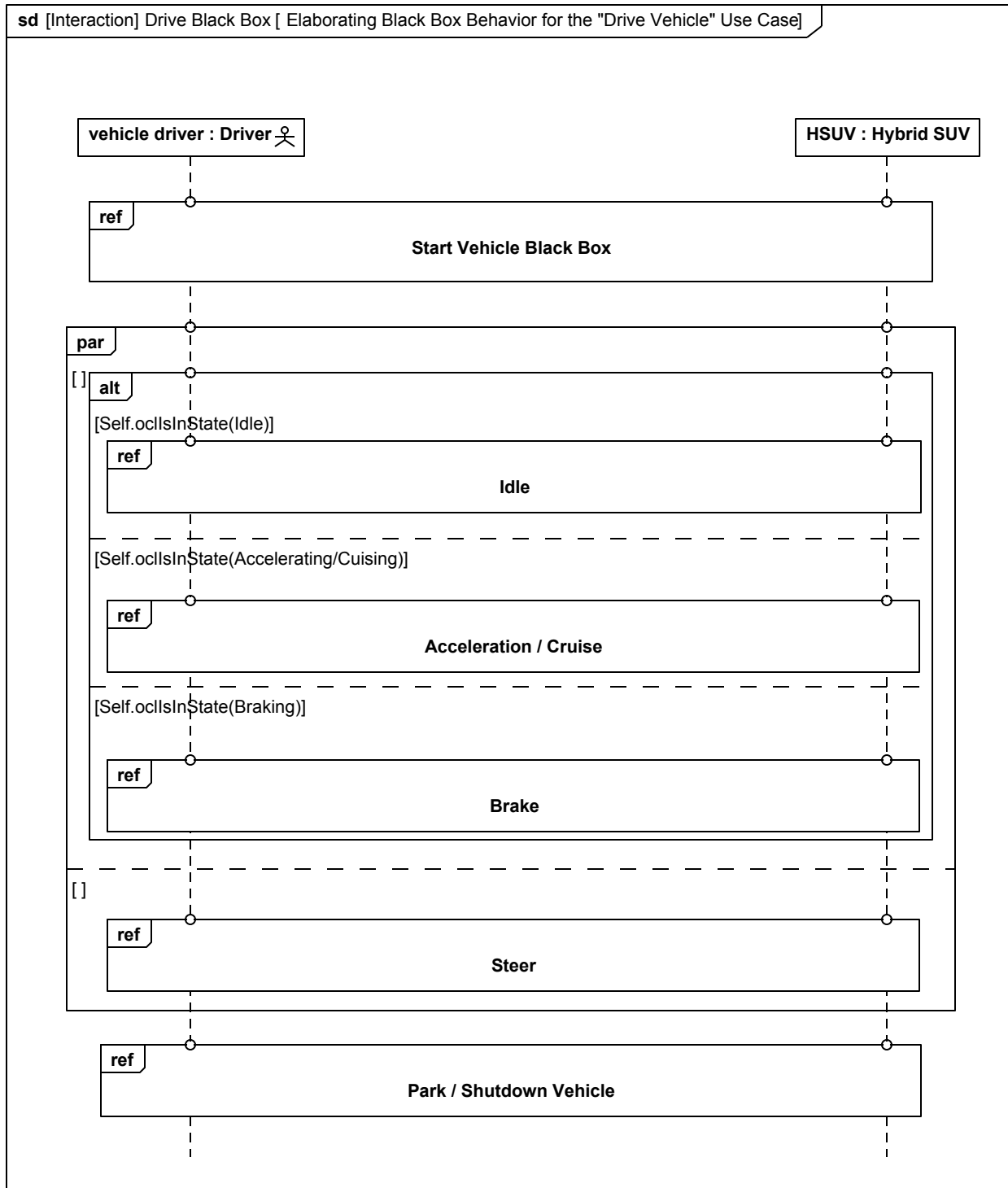


Figure D.7. Elaborating Black Box Behavior for the "Drive Vehicle" Use Case

D.4.3.2 State Machine Diagram - HSUV Operational States

Fig. D.8 depicts the operational states of the HSUV block, via a State Machine named “HSUVOperationalStates.” Note that this state machine was developed in conjunction with the DriveBlackBox interaction in Fig. D.7. Also note that this state machine refines the requirement “PowerSourceManagement,” which will be elaborated in the requirements sub clause of this sample problem. This diagram expresses only the nominal states. Exception states, like “acceleratorFailure,” are not expressed on this diagram.

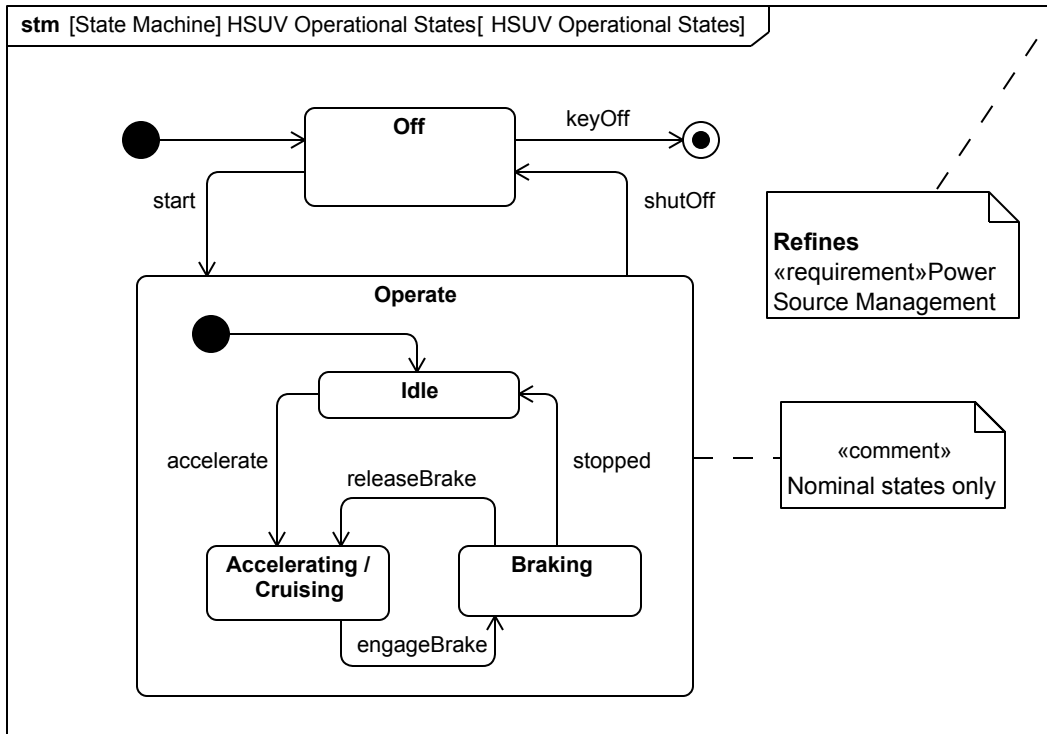


Figure D.8. HSUV Operational States

D.4.3.3 Sequence Diagram - Start Vehicle Black Box & White Box

Fig. D.9 shows a “black box” interaction, but references “Start Vehicle White Box” (Fig. D.10), which will decompose the lifelines within the context of the HybridSUV block.

The lifelines on Fig. D.10 (“whitebox” sequence diagram) need to come from the Power System decomposition. This now begins to consider parts contained in the HybridSUV block.

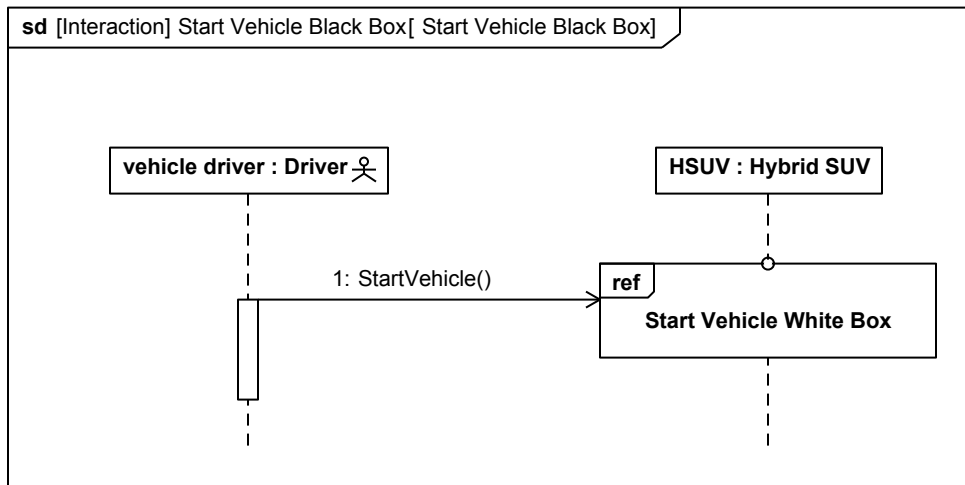


Figure D.9. Start Vehicle Black Box

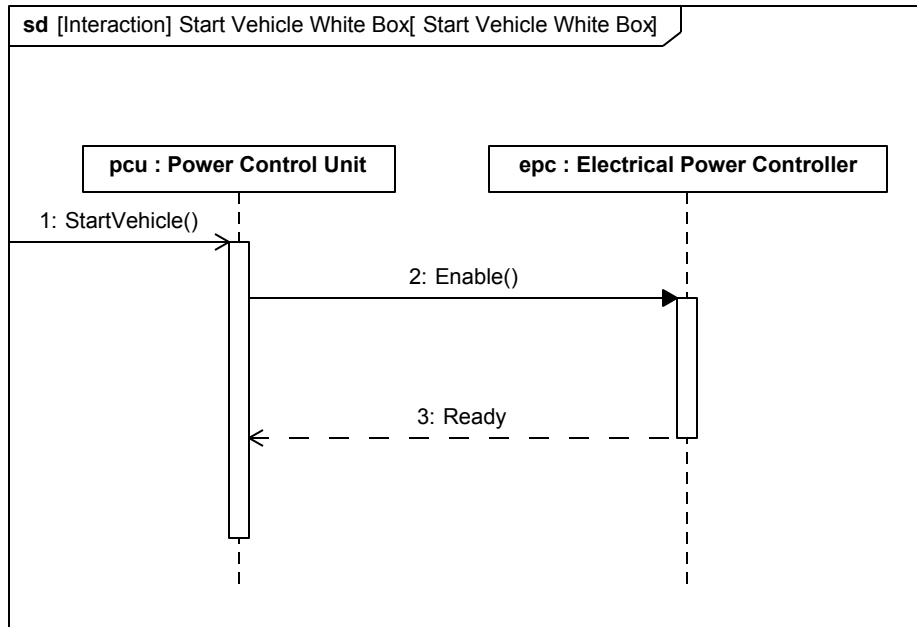


Figure D.10. Start Vehicle White Box

D.4.4 Establishing Requirements (Requirements Diagrams and Tables)

D.4.4.1 Requirement Diagram - HSUV Requirement Hierarchy

The vehicle system specification contains many text based requirements. A few requirements are highlighted in [Fig. D.11](#), including the requirement for the vehicle to pass emissions standards, which is expanded for illustration purposes. The containment (cross hair) relationship between requirements, for purpose of this example, refers to the practice of decomposing a complex requirement into simpler, single requirements.

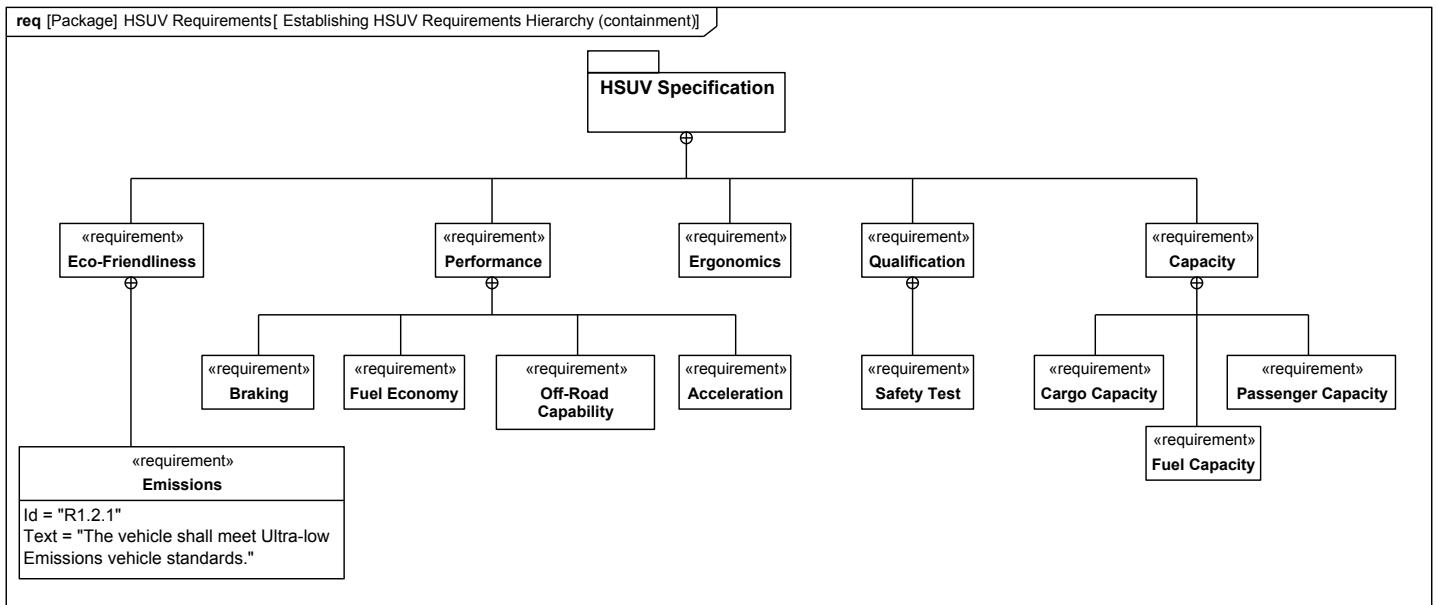


Figure D.11. Establishing HSUV Requirements Hierarchy (containment) - (Requirements Diagram)

D.4.4.2 Requirement Diagram - Derived Requirements

[Fig. D.12](#) shows a set of requirements derived from the lowest tier requirements in the HSUV specification. Derived requirements, for the purpose of this example, express the concepts of requirements in the HSUV Specification in a manner that specifically relates them to the HSUV system. Various other model elements may be necessary to help develop a derived requirement, and these model element may be related by a «refinedBy» relationship. Note how Power Source Management is “RefinedBy” the HSUV Operational States model ([Fig. D.8](#)). Note also that rationale can be attached to the «deriveReq» relationship. In this case, rationale is provided by a referenced document “Hybrid Design Guidance.”

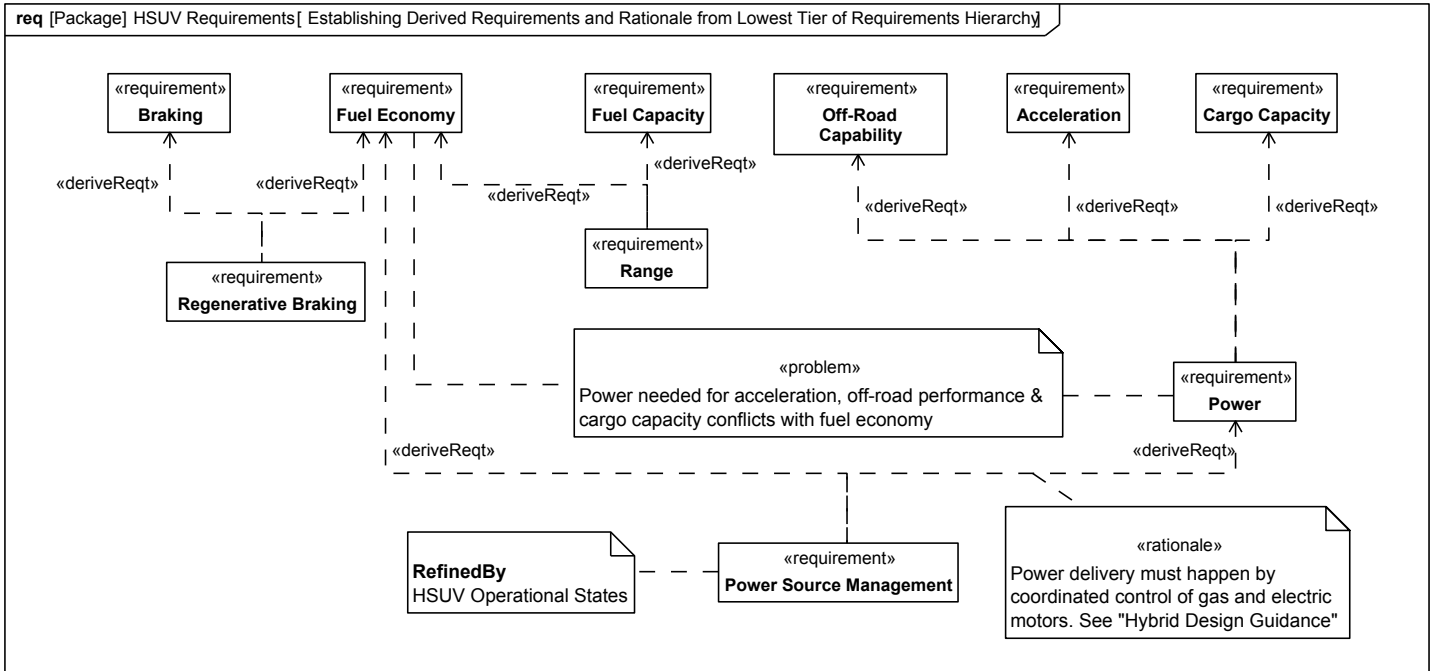


Figure D.12. Establishing Derived Requirements and Rationale from Lowest Tier of Requirements Hierarchy (Requirements Diagram)

D.4.4.3 Requirement Diagram - Acceleration Requirement Relationships

Section D.4.4.2 focuses on the Acceleration requirement, and relates it to other requirements and model elements. The “refine” relation, introduced in Fig. D.12, shows how the Acceleration requirement is refined by a similarly named use case. The Power requirement is satisfied by the part property pwr-ss:Power Subsystem, and a Max Acceleration test case verifies the Acceleration requirement.

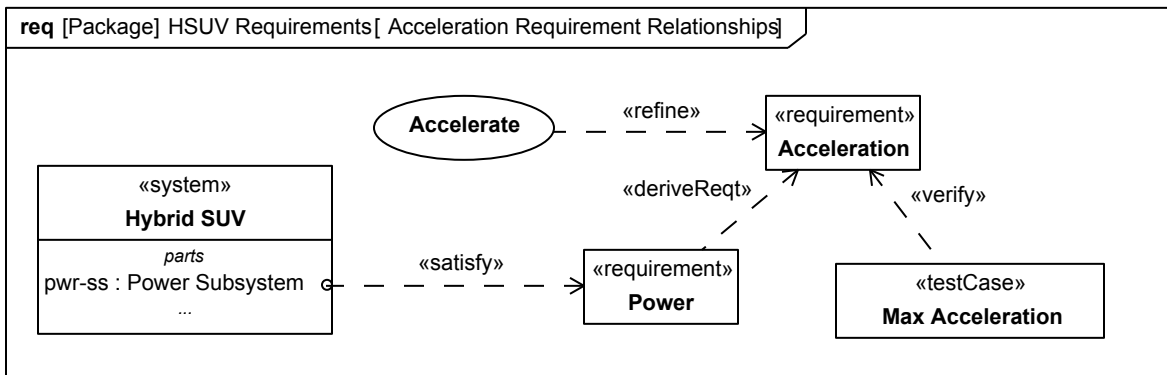


Figure D.13. Acceleration Requirement Relationships (Requirements Diagram)

D.4.4.4 Table - Requirements Table

SysML allows the representation of relationships using tables without constraining the exact layout of such a table.

Section D.4.4.4 provides two examples showing requirement containment (decomposition), and requirements derivation in tabular form. This is a more compact representation than the requirements diagrams shown previously.

#	Id	△ Name	Text
1	2	⊖ R 2 Performance	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically better fuel economy.
2	2.1	R 2.1 Acceleration	The Hybrid SUV shall have the braking capability of a typical SUV.
3	2.2	R 2.2 Braking	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
4	2.3	R 2.3 Fuel Economy	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
5	2.4	R 2.4 Off-Road Capability	The Hybrid SUV shall have the off-road capability of a typical SUV.

#	△ Id	Name	R1 Id	Derived	R2 Id	Derived 2
1	2.1	R Acceleration	d.4	R Power	d.2	R Power Source Management
2	2.2	R Braking	d.1	R Regenerative Braking		
3	2.3	R Fuel Economy	d.2	R Power Source Management		
			d.1	R Regenerative Braking		
			d.3	R Range		
4	2.4	R Off-Road Capability	d.4	R Power	d.2	R Power Source Management
5	4.1	R Cargo Capacity	d.4	R Power	d.2	R Power Source Management

Figure D.14. Requirements Relationships Expressed in Tabular Format

Please Note: Tables are not a normative part of the SysML specification. So all tables are for illustration and reference only.

D.4.5 Breaking Down the Pieces (Block Definition Diagrams, Internal Block Diagrams)

D.4.5.1 Block Definition Diagram - Automotive Domain

Fig. D.15 provides definition for the concepts previously shown in the context diagram. Note that the interactions Drive Black Box and Start Vehicle Black Box (described in, Fig. D.9 and Fig. D.10) are depicted as owned by the AutomotiveDomain block.

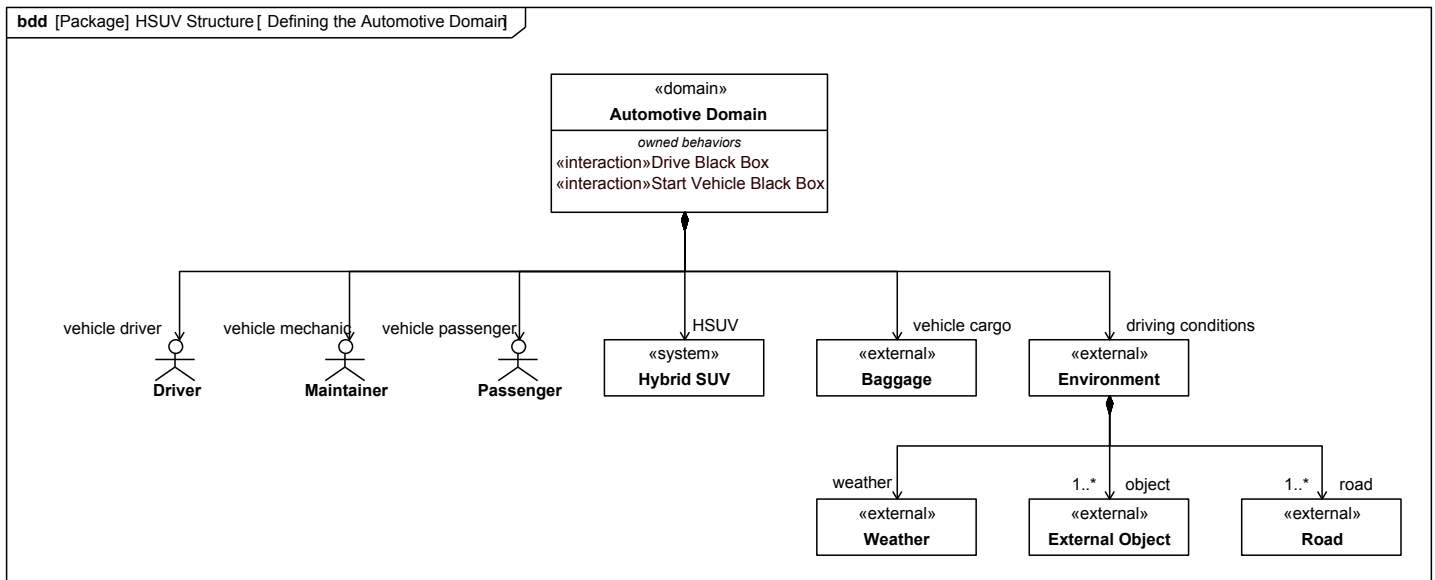


Figure D.15. Defining the Automotive Domain

D.4.5.2 Block Definition Diagram - Hybrid SUV

Fig. D.16 defines components of the Hybrid SUV block. Note that the Brake Pedal and Wheel Hub Assembly are used by, but not contained in, the Power Subsystem block.

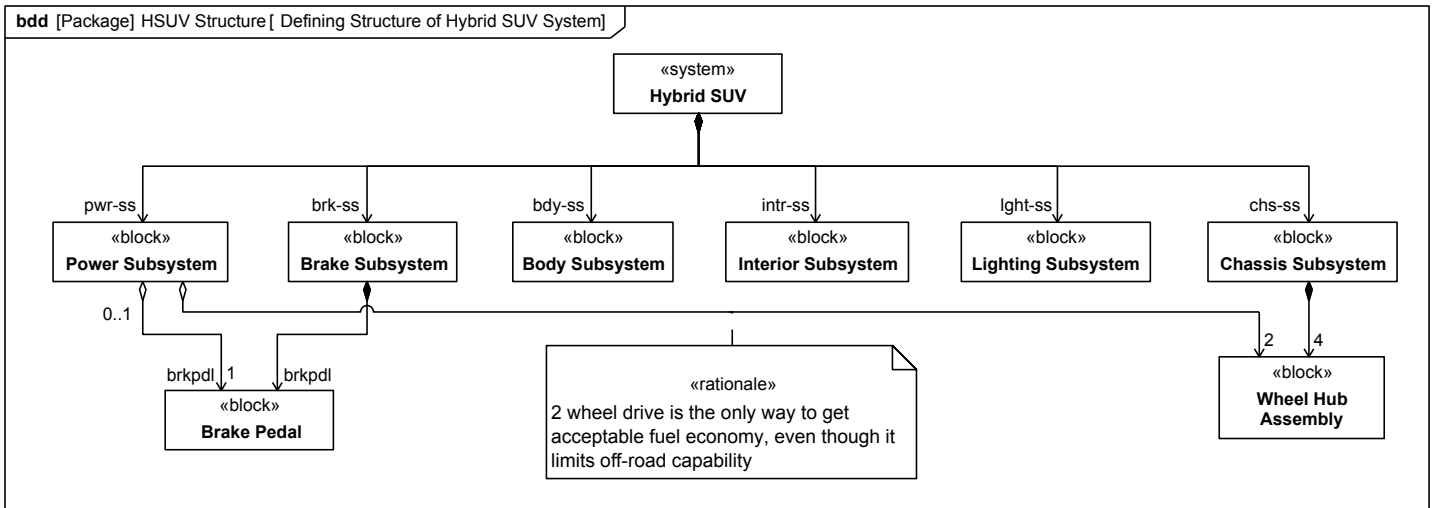


Figure D.16. Defining Structure of Hybrid SUV System

D.4.5.3 Internal Block Diagram - Hybrid SUV

Fig. D.17 shows how the top level model elements in the above diagram are connected together in the Hybrid SUV block.

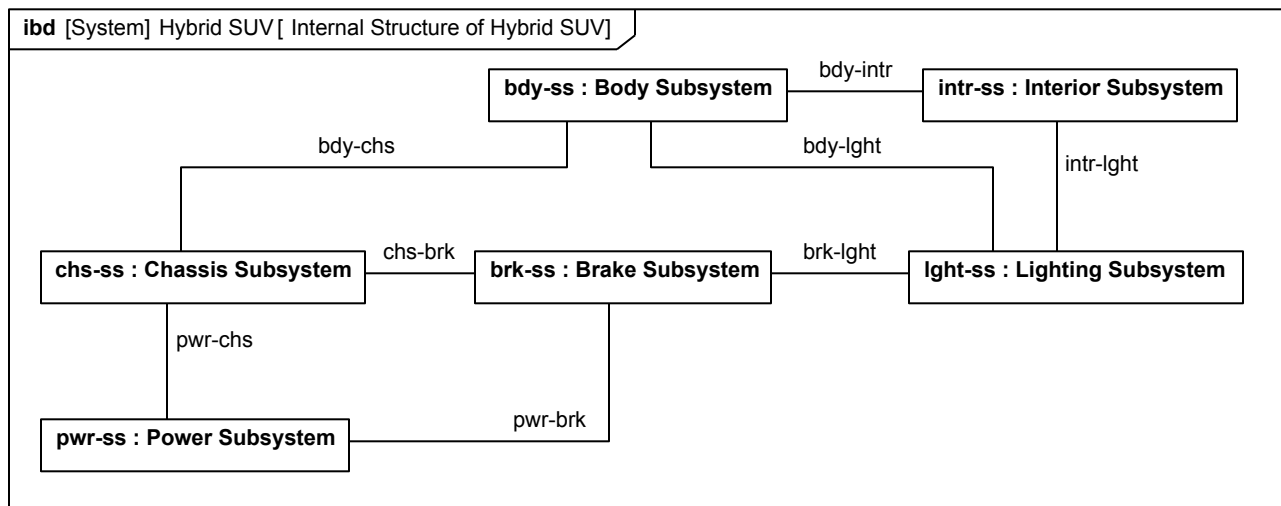


Figure D.17. Internal Structure of Hybrid SUV (Internal Block Diagram)

D.4.5.4 Block Definition Diagram - Power Subsystem

Fig. D.18 defines the next level of decomposition, namely the components of the Power Subsystem block. Note how the use of white diamond (shared aggregation) on Front Wheel, Brake Pedal, and others denotes the same “use-not- composition” kind of relationship previously shown in Fig. D.16

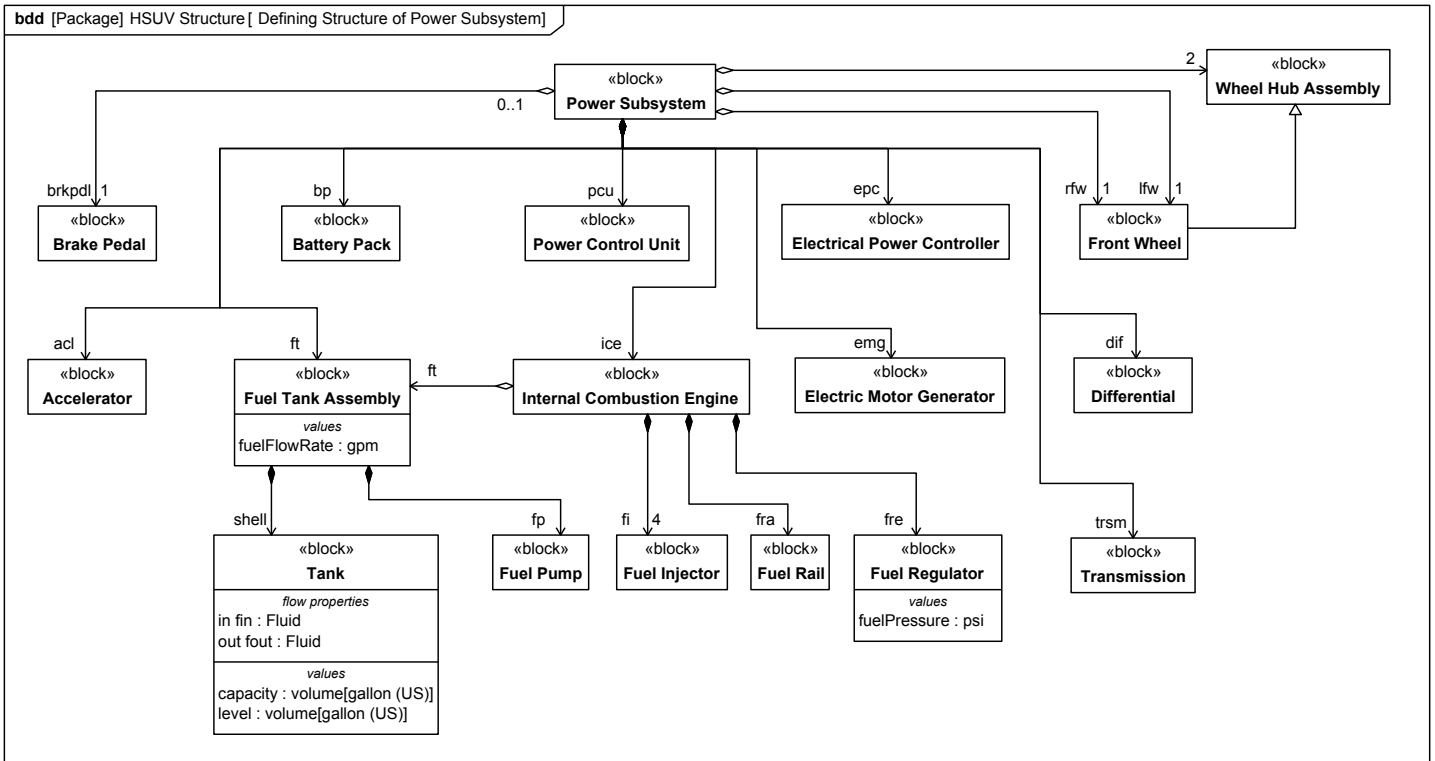


Figure D.18. Defining Structure of Power Subsystem

D.4.5.5 Internal Block Diagram for the "Power Subsystem"

Fig. D.19 shows how the parts of the Power Subsystem block, as defined in the diagram above, are used. It shows connectors between parts, ports, and connectors with item flows. The dashed borders on Front Wheel and Brake Pedal denote the “use-not-composition” relationship depicted elsewhere in Fig. D.16 and Fig. D.18.

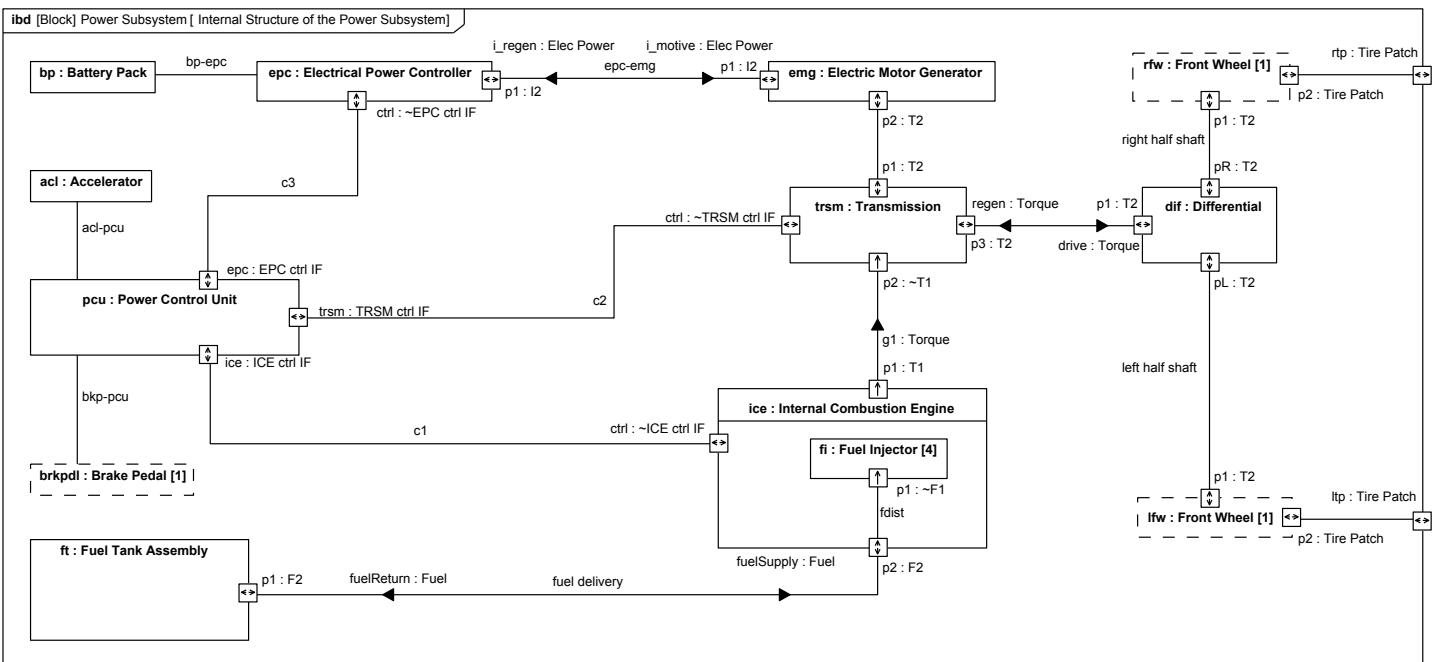


Figure D.19. Internal Structure of the Power Subsystem

Fig. D.20 provides definition of the block that types the ports linked by connectors c1, c2 and c3 in Fig. D.19. Note the use of conjugate (~) interface blocks, the composition of interface blocks, and the use of signals & value types for flow properties.

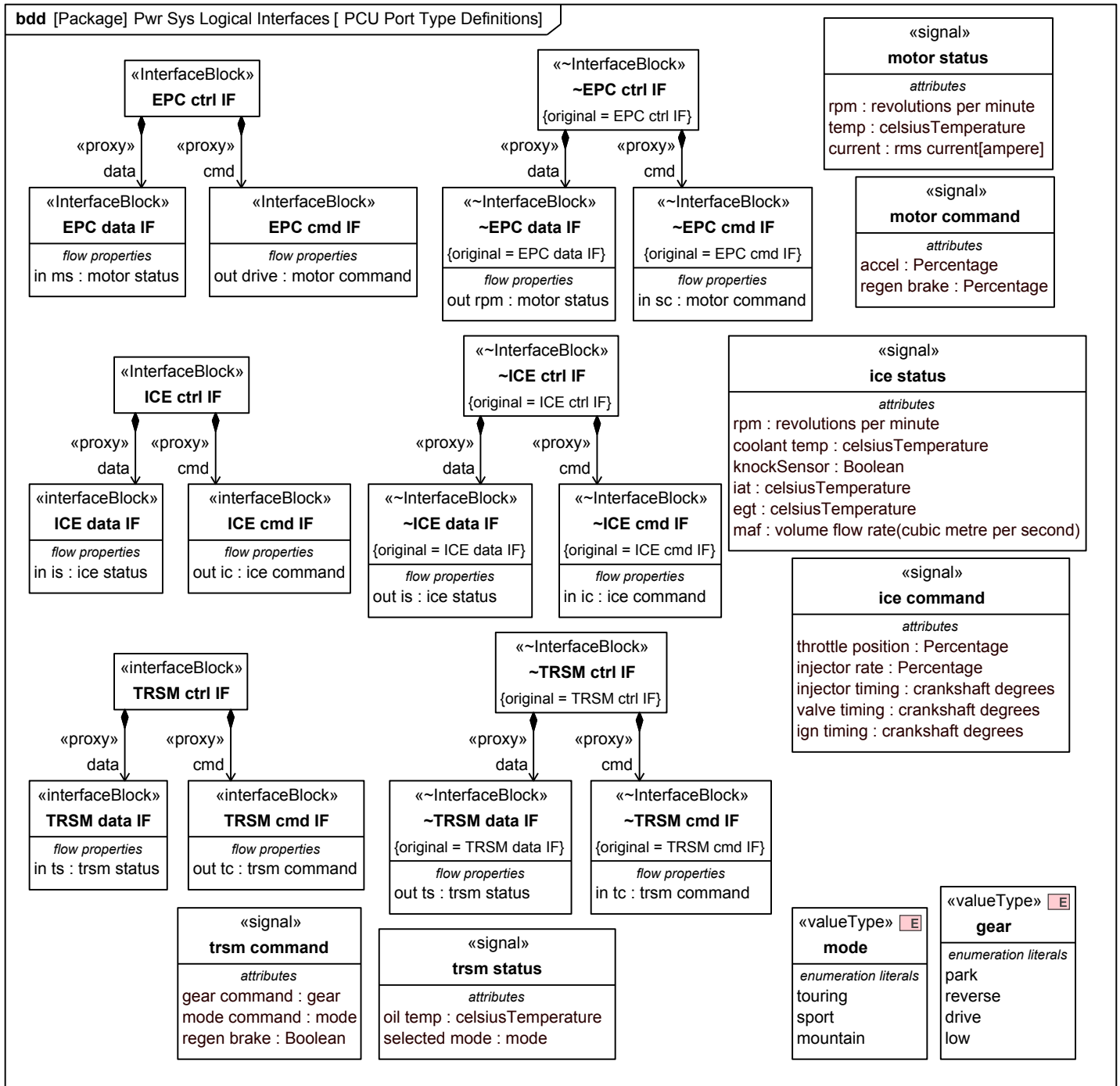


Figure D.20. PCU Port Type Definitions

D.4.6 Defining Ports and Flows

D.4.6.1 Block Definition Diagram - ICE Flow Properties

For purpose of example, the ports, flows, and related point-to-point connectors in [Fig. D.19](#) are being refined into a common bus architecture. For this example, ports with flow properties have been used to model the bus architecture. [Fig. D.21](#) is an incomplete first step in the refinement of this bus architecture, as it begins to specify the flow properties for Internal Combustion Engine, the Transmission, and the Electrical Power Controller.

Note that the table provided is not a SysML diagram, but is a quite useful list of CAN Message interface specifications depicting various messages that may be used on the CAN bus.

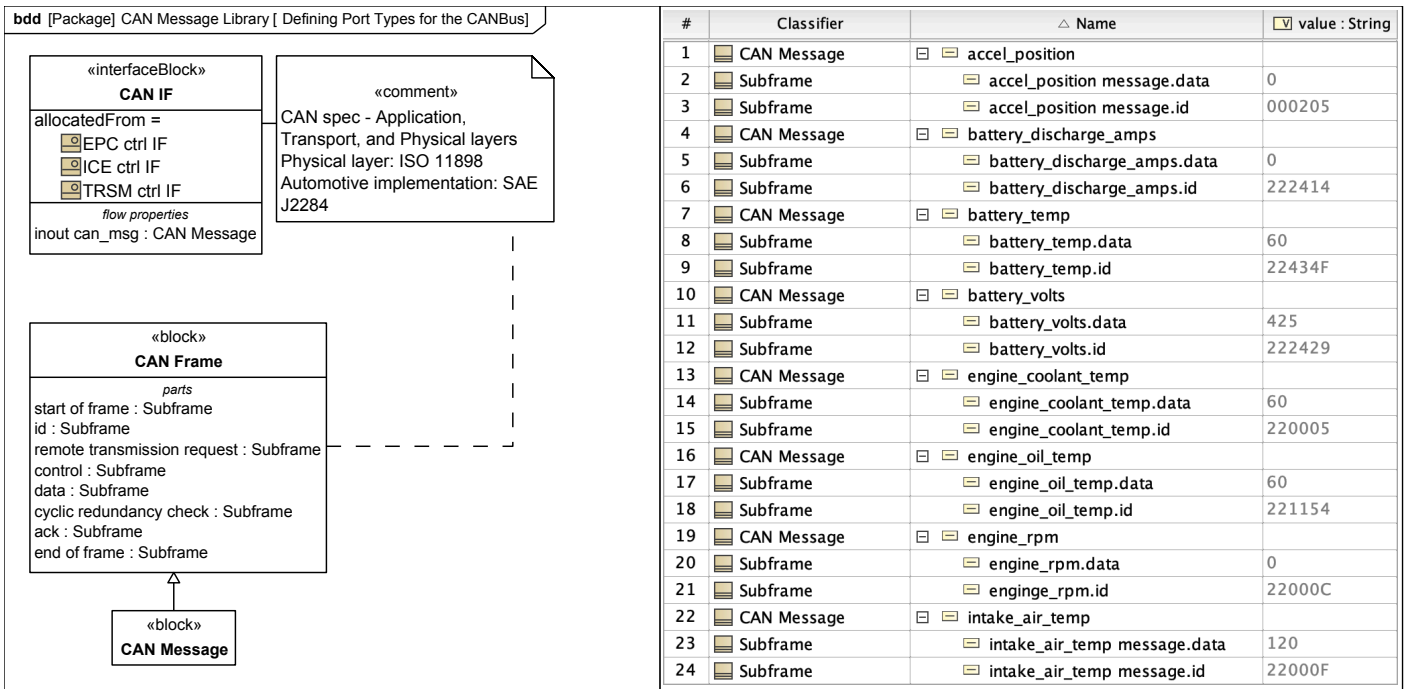


Figure D.21. Initially Port Types with Flow Properties for the CANBus (Block Definition Diagram)

Please Note: Tables are not a normative part of the SysML specification. So all tables are for illustration and reference only.

D.4.6.2 Internal Block Diagram - CANbus

Fig. D.22 continues the refinement of this Controller Area Network (CAN) bus architecture using ports. The explicit structural allocation between the original connectors of Fig. D.19 and this new bus architecture is shown in Fig. D.39.

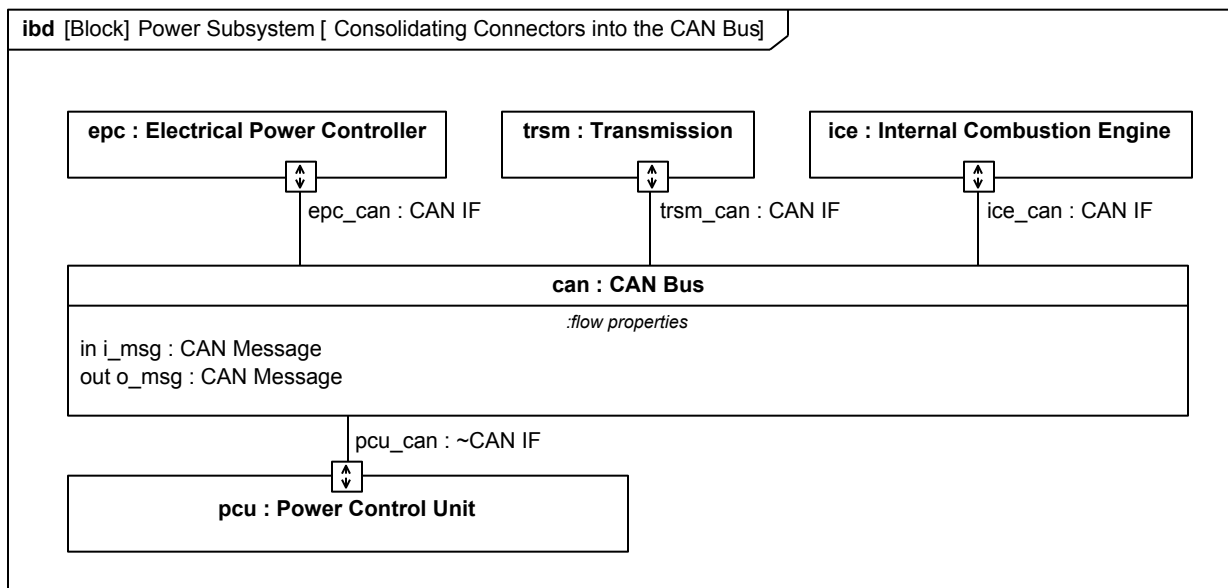


Figure D.22. Consolidating Connectors into the CAN Bus

D.4.6.3 Block Definition Diagram - Fuel Flow Properties

The ports on the Fuel Tank Assembly and Internal Combustion Engine (as shown in Fig. D.19) are defined in Fig. D.23.

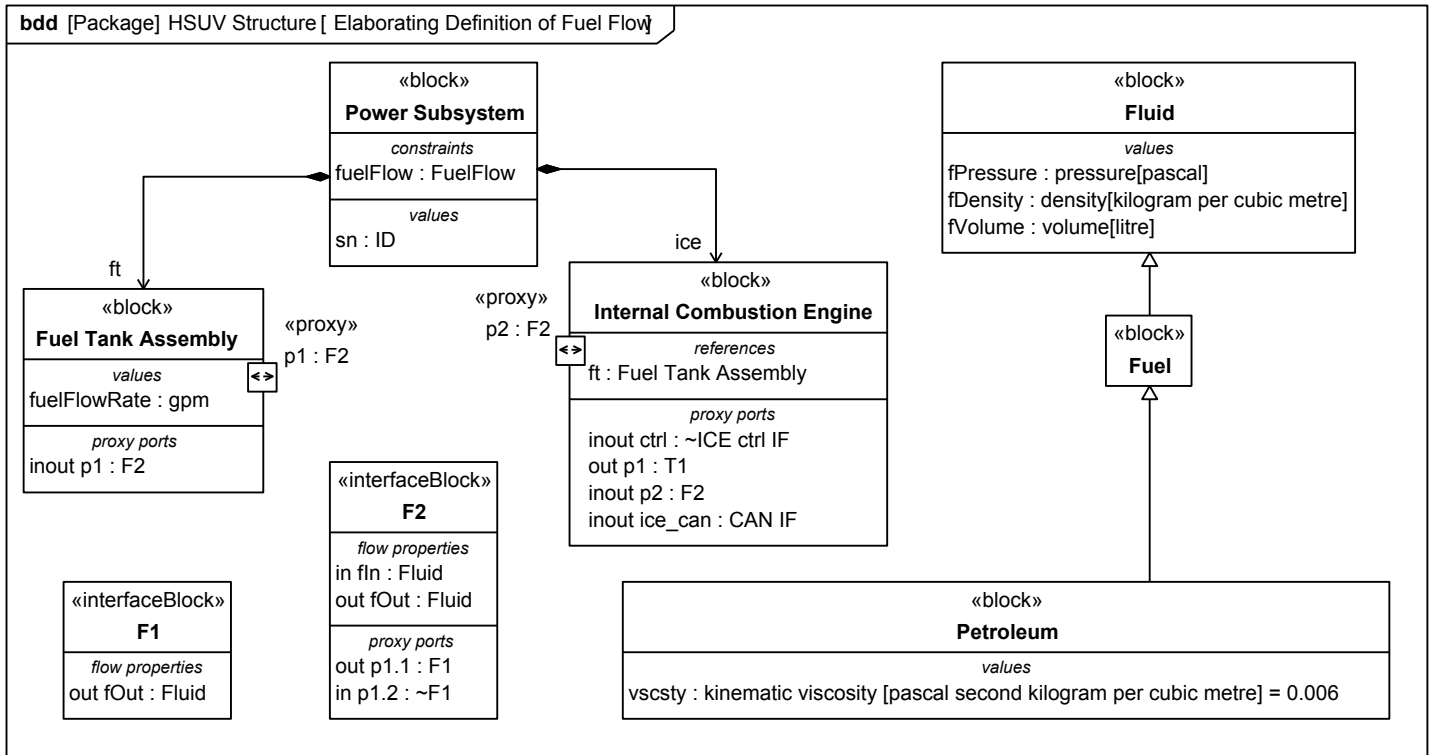


Figure D.23. Elaborating Definition of Fuel Flow

D.4.6.4 Parametric Diagram - Fuel Flow

Fig. D.24 is a parametric diagram showing how fuel flow rate is related to FuelDemand and FuelPressure value properties.

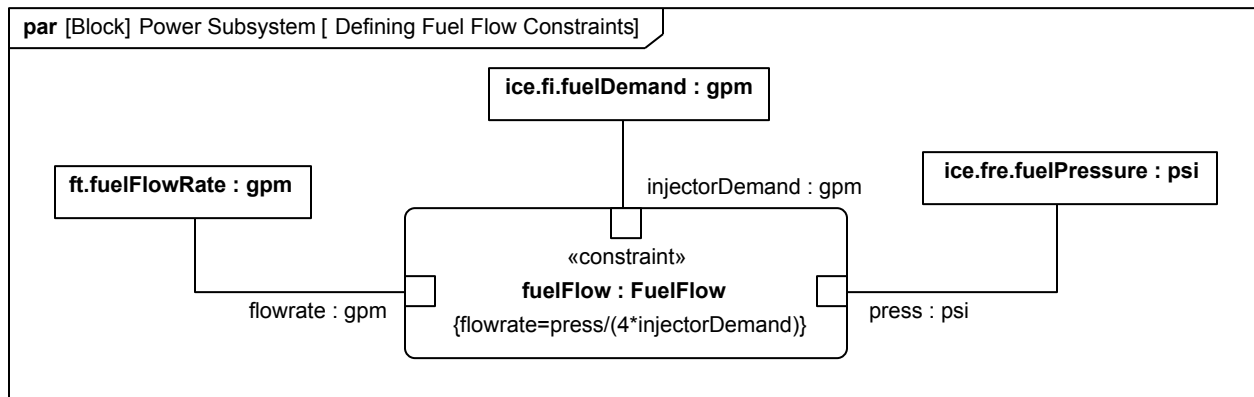


Figure D.24. Defining Fuel Flow Constraints

D.4.6.5 Internal Block Diagram - Fuel Distribution

Fig. D.25 shows how the connectors fuelDelivery and fdist on Fig. D.19 have been expanded to include design detail. The fuel delivery connector is allocated to two connectors, one carrying fuel supply and the other carrying fuel return. The fdist connector inside the Internal Combustion Engine block has been allocated into the fuel regulator and fuel rail parts. These more detailed design elements are related to the original connectors using the allocation relationship. Fuel in the tank portion of the Fuel Tank Assembly is drawn by the Fuel Pump for use in the engine, and is refreshed, to some degree, by fuel returning to the Fuel Tank Assembly.

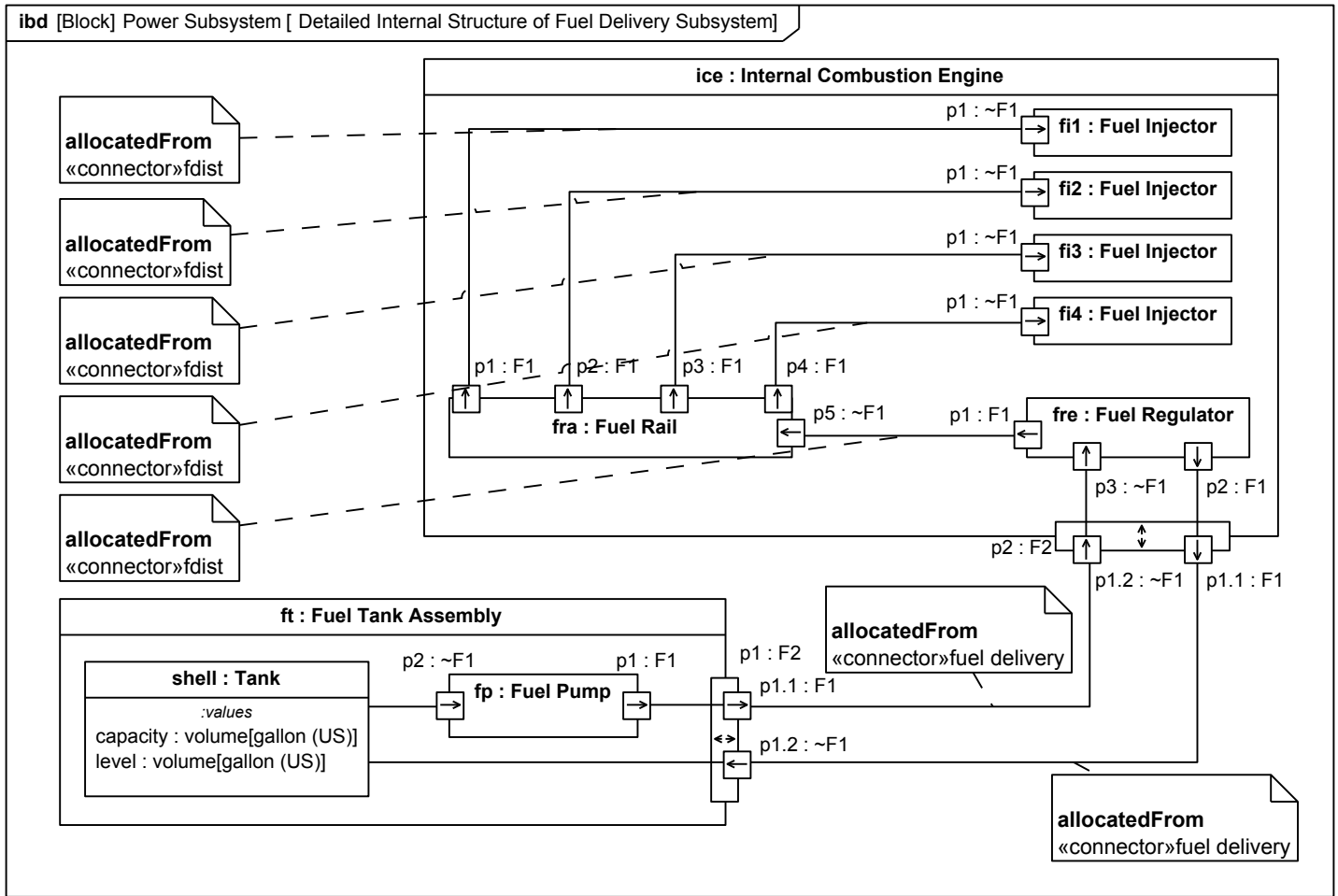


Figure D.25. Detailed Internal Structure of Fuel Delivery Subsystem (Internal Block Diagram)

D.4.7 Analyze Performance (Constraint Diagrams, Timing Diagrams, Views)

D.4.7.1 Block Definition Diagram - Analysis Context

Fig. D.26 defines the various model elements that will be used to conduct analysis in this example. It depicts each of the constraint blocks/equations that will be used for the analysis, and key relationships between them. The types of the constraint parameters have defaulted to Real, but will need to be updated to the actual value types of the properties to which they are bound.

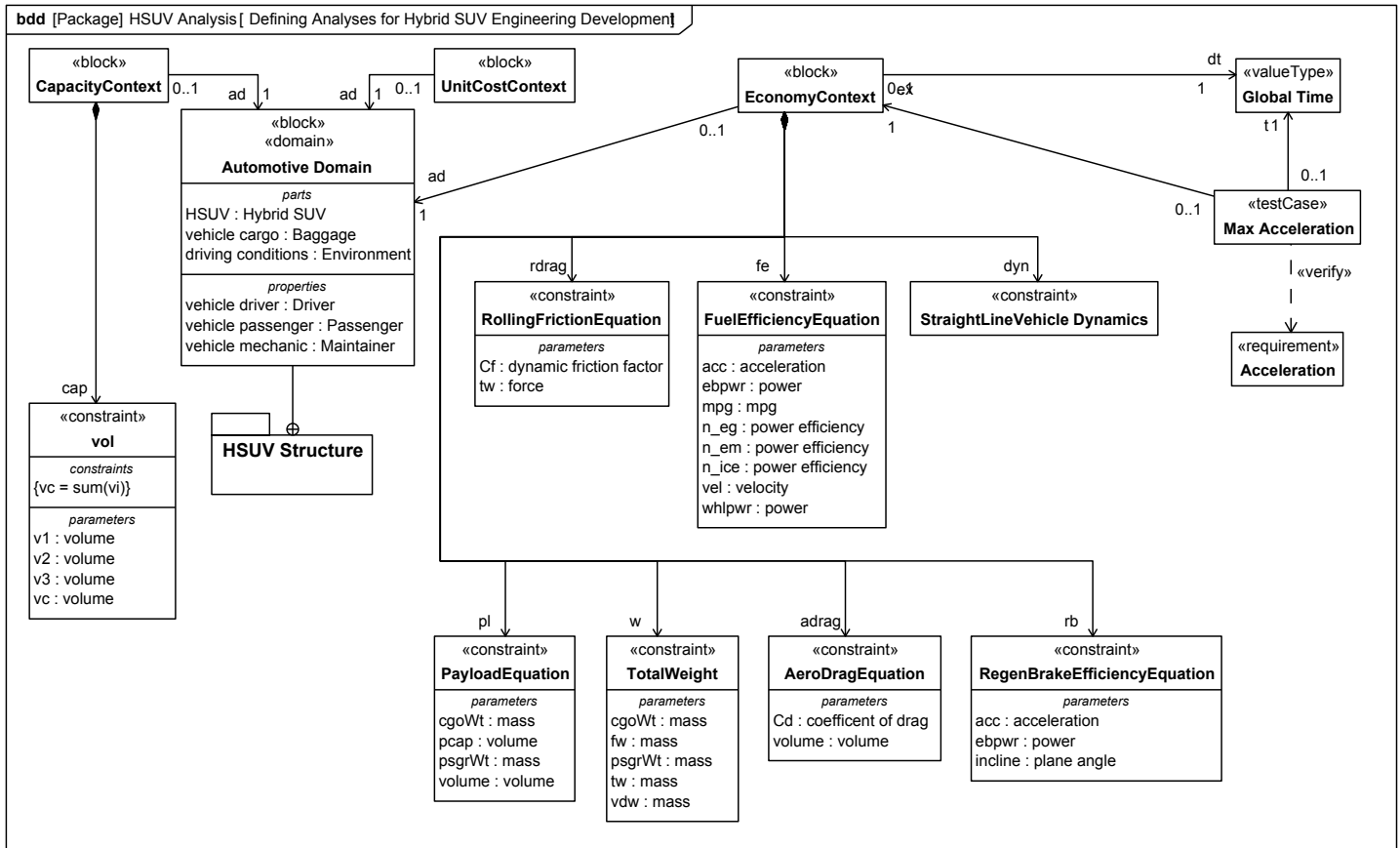


Figure D.26. Defining Analyses for Hybrid SUV Engineering Development

D.4.7.2 Package Diagram - Performance View Definition

Fig. D.27 shows the user-defined Performance Viewpoint, and the elements that populate the HSUV specific Hybrid SUV Performance View. This view may contain a number of other views, as well as exposing specific model elements or package contents.

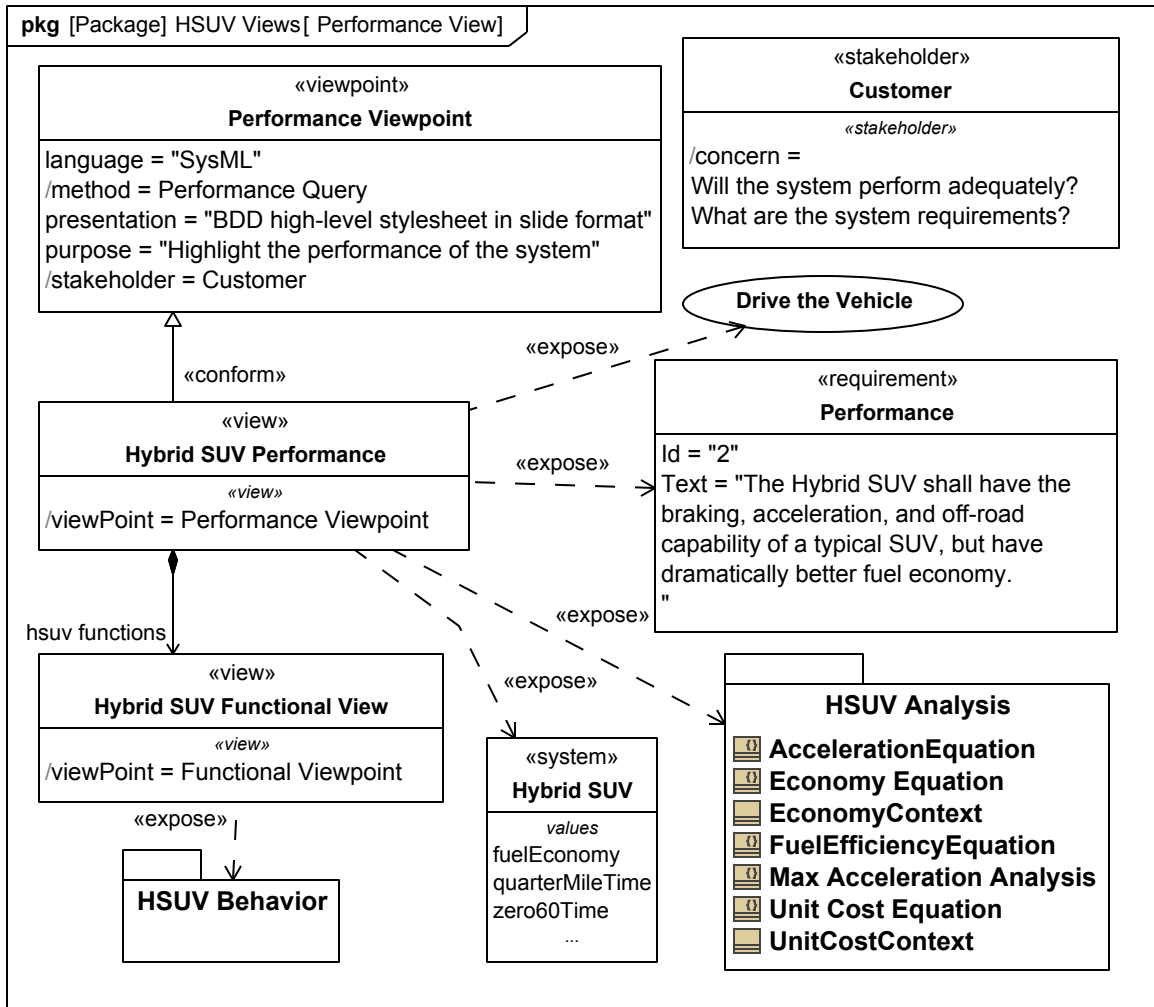


Figure D.27. Performance View

D.4.7.3 Package Diagram - Viewpoint Definition

Fig. D.28 shows the Requirements and VnV viewpoint definitions and their conforming views. The Customer stakeholder is referenced by both viewpoints and both views.

Note that the value of the stakeholder property is an instance of the stereotype not the class to which the stereotype is applied.

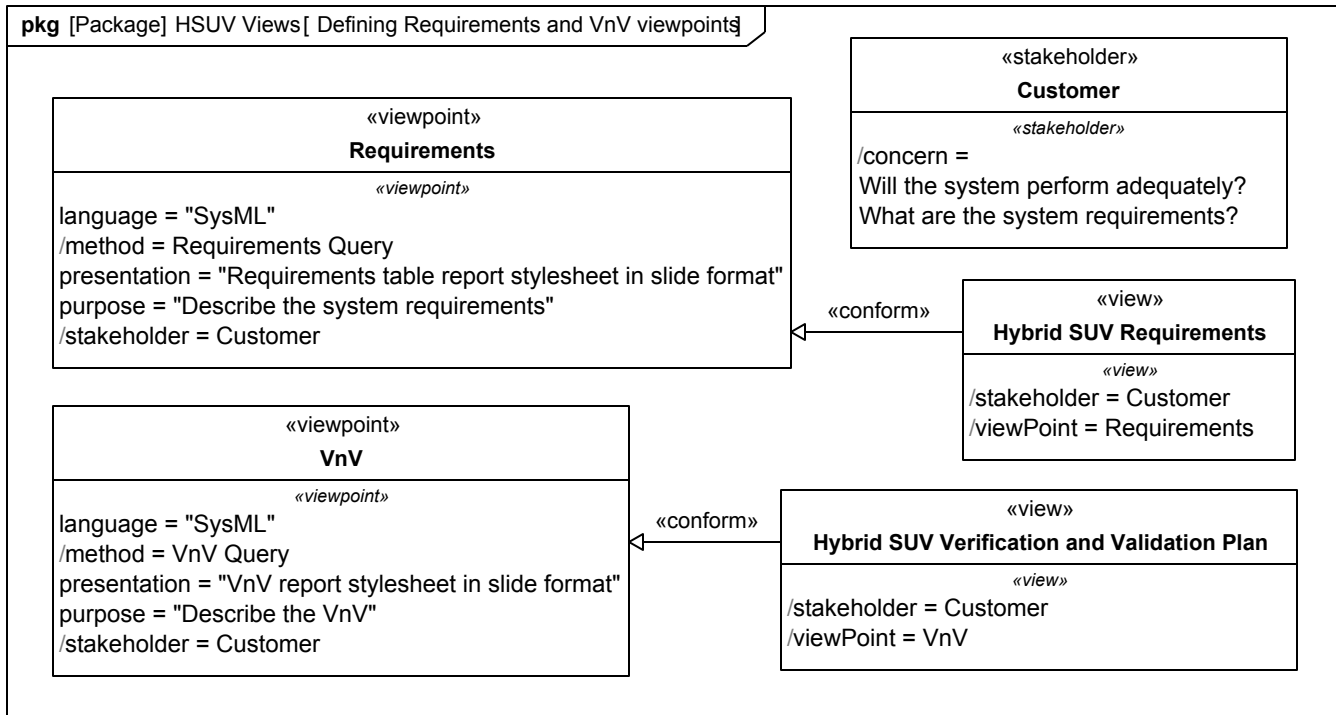


Figure D.28. Defining Requirements and VnV viewpoints

D.4.7.4 Package Diagram - View Definition

Fig. D.29 shows the Requirements and VnV views and the model elements they expose. Note that the expose relationship relies on the viewpoint method to identify the entire set of elements that appear in the view.

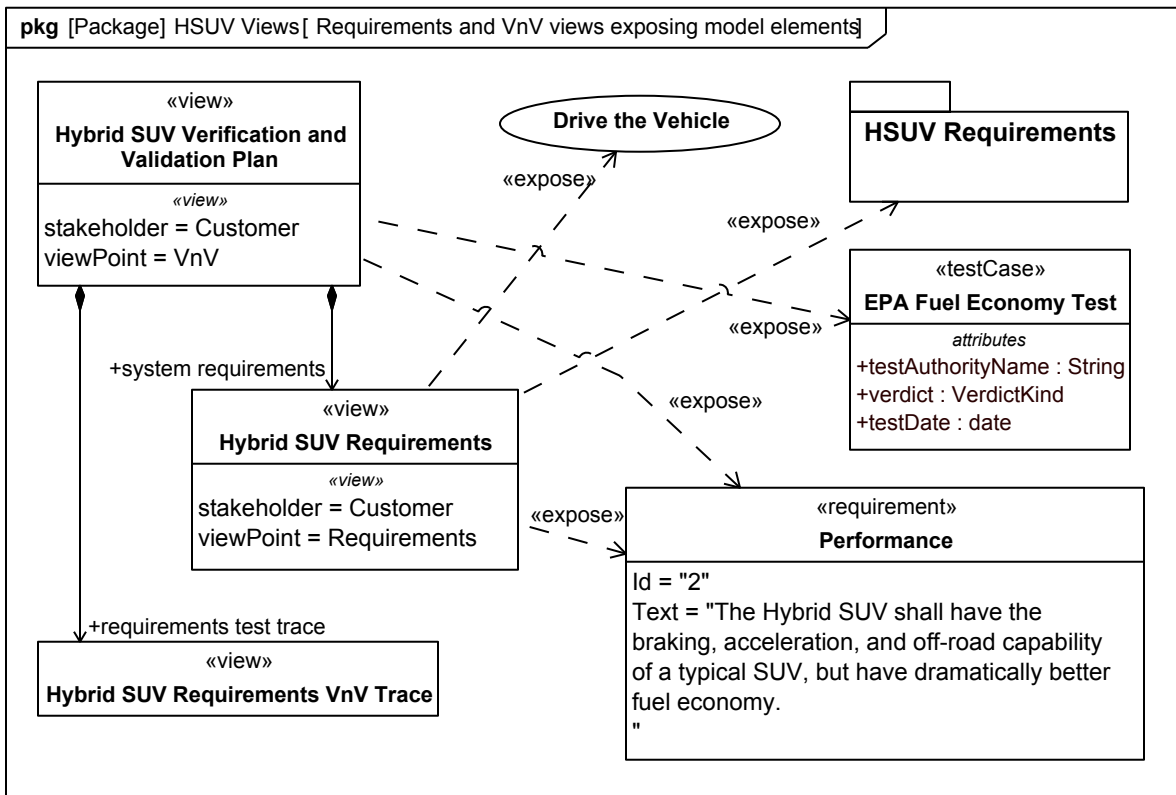


Figure D.29. Requirements and VnV views exposing model elements

D.4.7.5 Package Diagram - View Hierarchy

Fig. D.30 shows the composition Hybrid SUV Verification and Validation Plan view and supporting views.

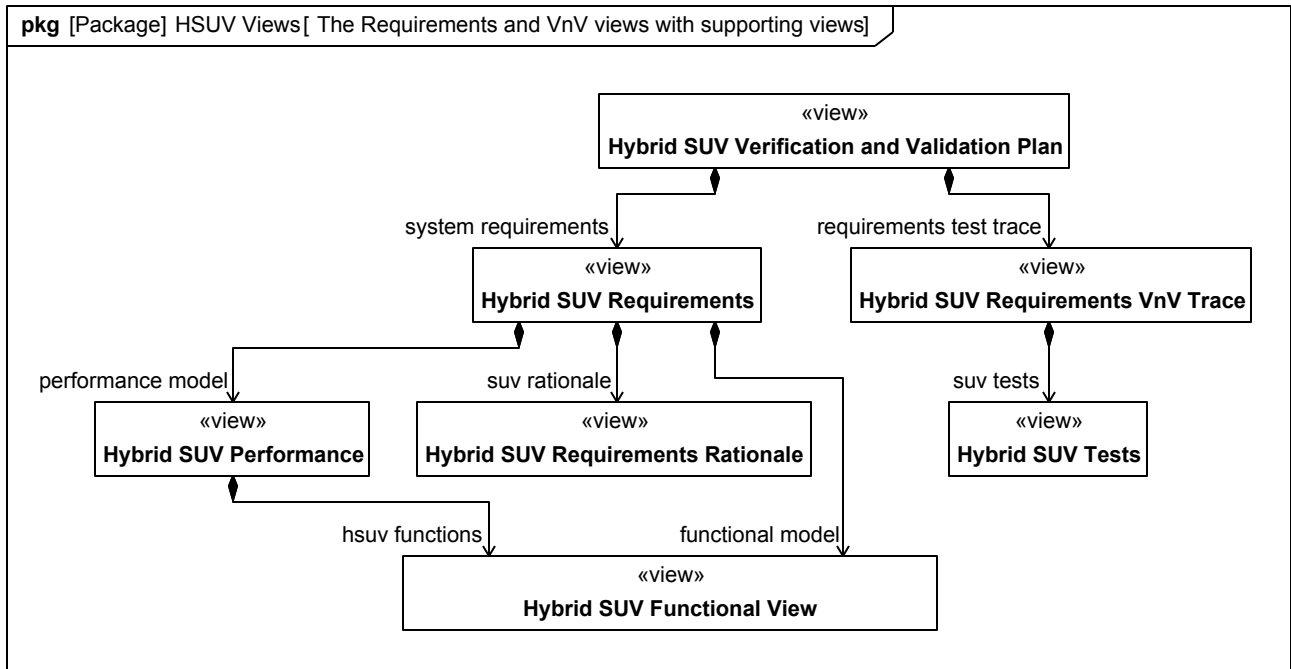


Figure D.30. The Requirements and VnV views with supporting views

D.4.7.6 Parametric Diagram - Measures of Effectiveness

Measure of Effectiveness is a user defined stereotype. [Fig. D.31](#) shows how the overall cost effectiveness of the HSUV will be evaluated. It shows the particular measures of effectiveness for one particular alternative for the HSUV design, and can be reused to evaluate other alternatives. Value types for the moe value properties are not shown on this diagram.

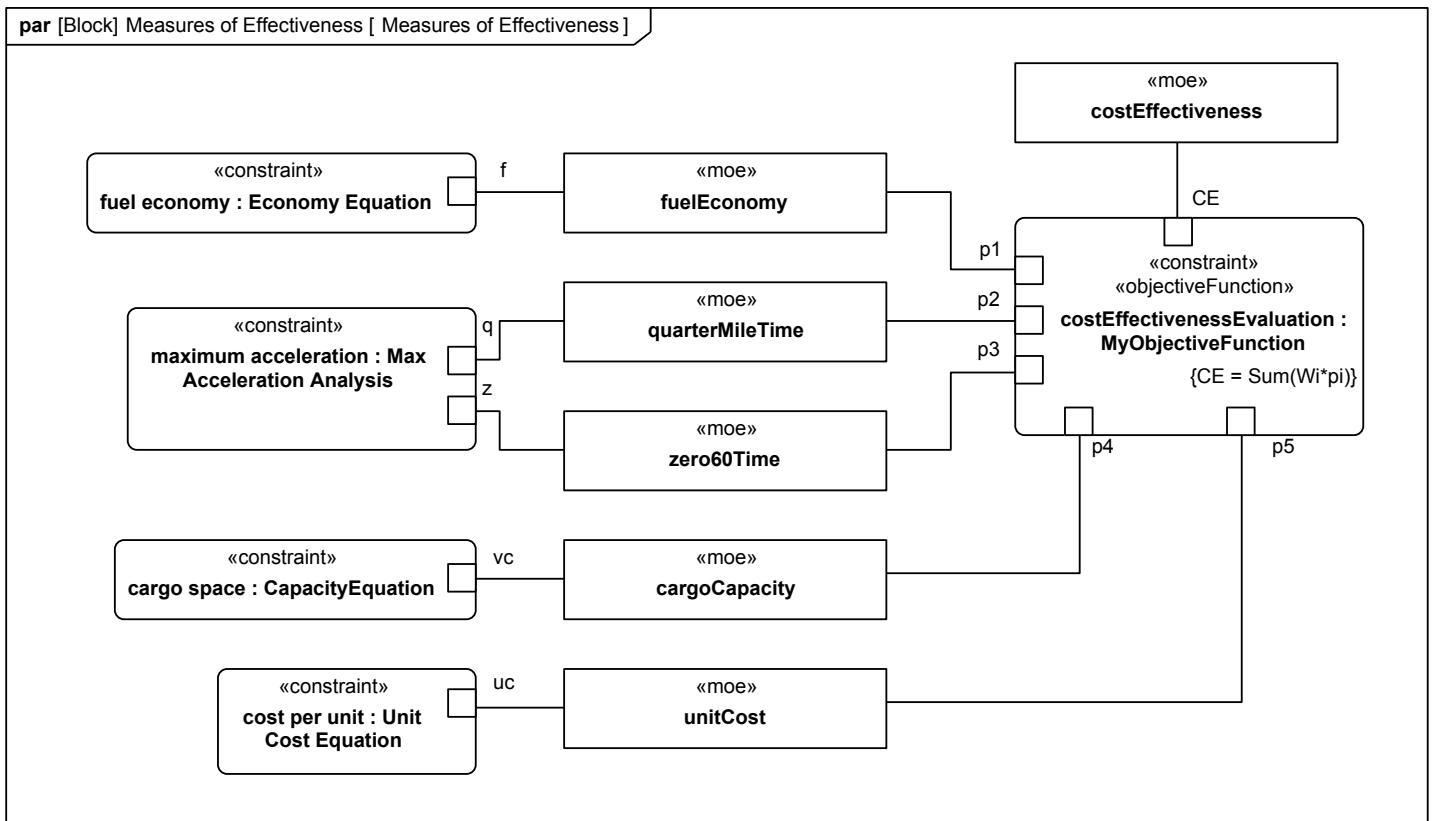


Figure D.31. Measures of Effectiveness

D.4.7.7 Parametric Diagram - Economy

Since overall fuel economy is a key requirement on the HSUV design, this example applies significant detail in assessing it. Fig. D.32 shows the constraint blocks and properties necessary to evaluate fuel economy. Value types for the value properties are not shown on this diagram.

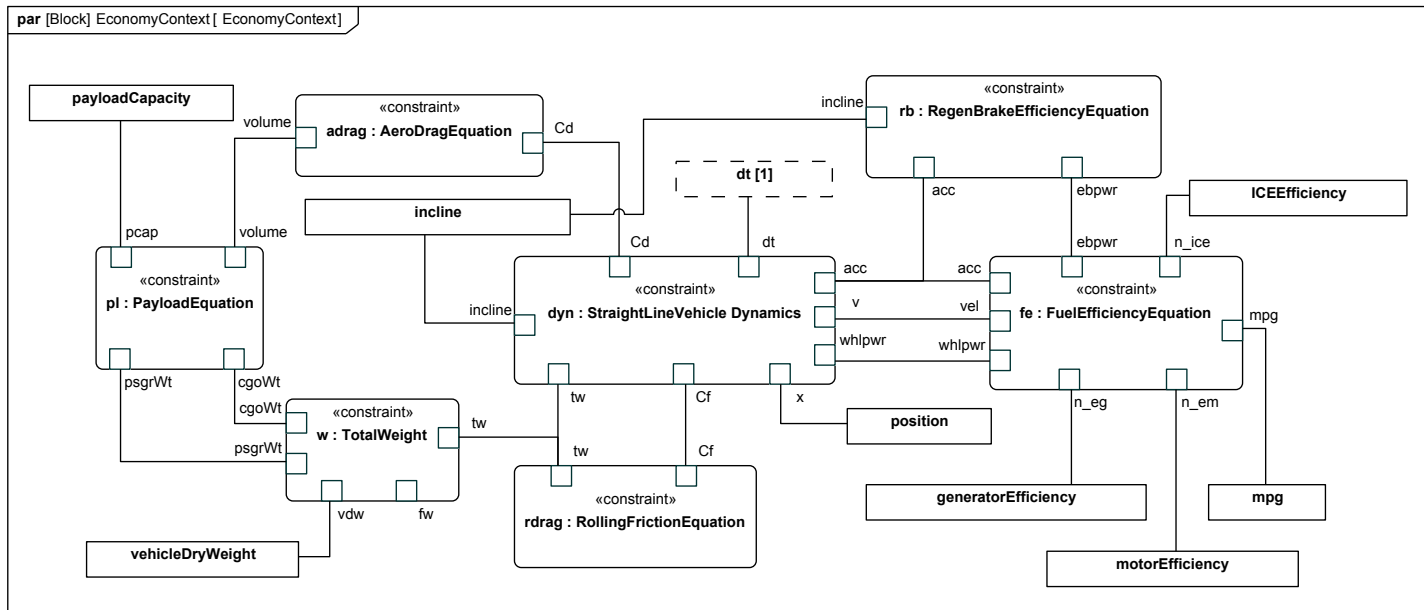


Figure D.32. EconomyContext

D.4.7.8 Parametric Diagram - Dynamics

The StraightLineVehicleDynamics constraint block from Fig. D.32 has been expanded in Fig. D.33. Each constraint is identified using curly brackets {}. In addition, Rationale has been used to explain the meaning of each constraint maintained.

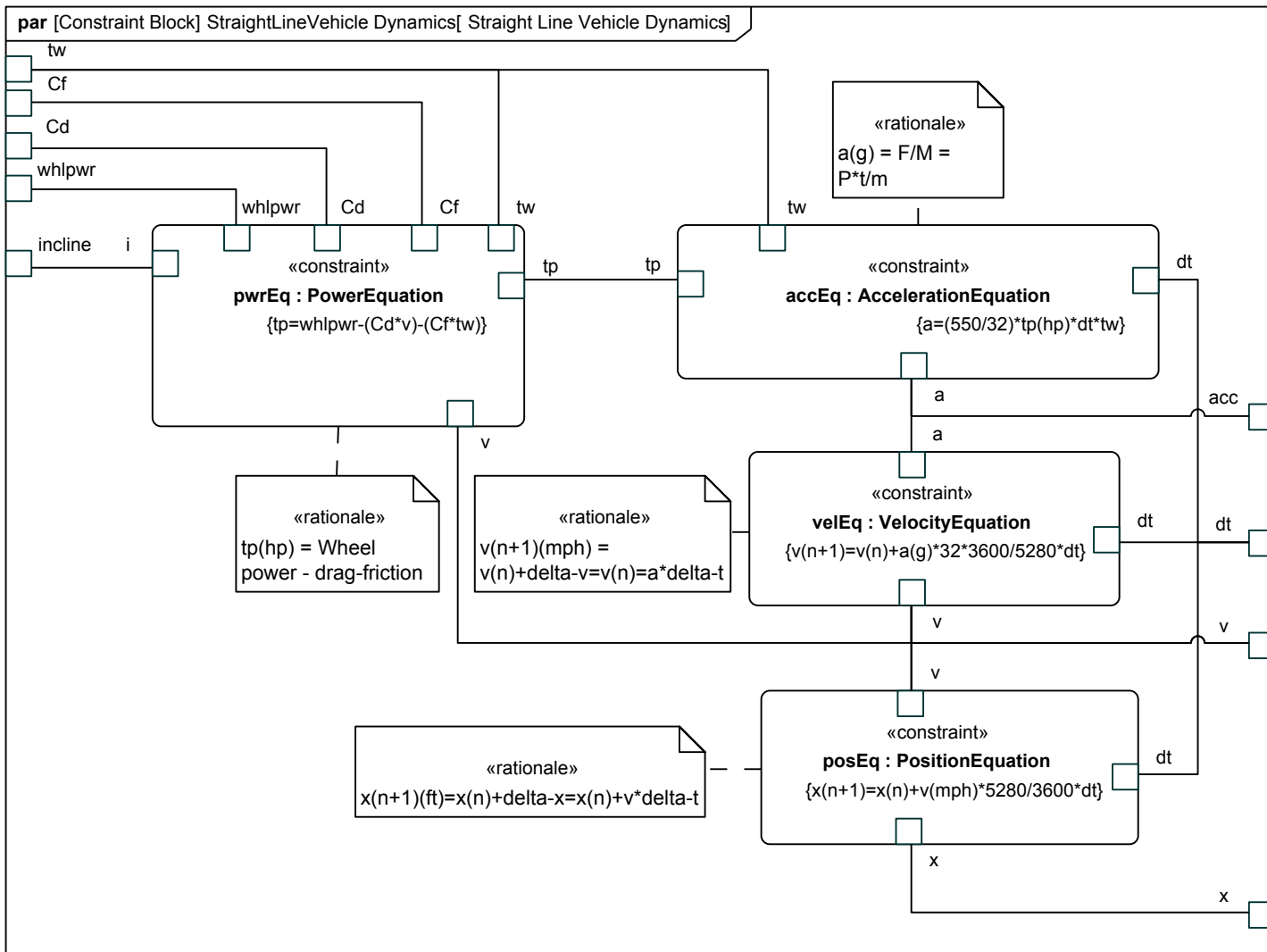


Figure D.33. Straight Line Vehicle Dynamics

The constraints and parameters in [Fig. D.33](#) are detailed in [Section D.4.7.8](#) in Block Definition Diagram format.

Note the use of valueTypes originally defined in [Fig. D.2](#).

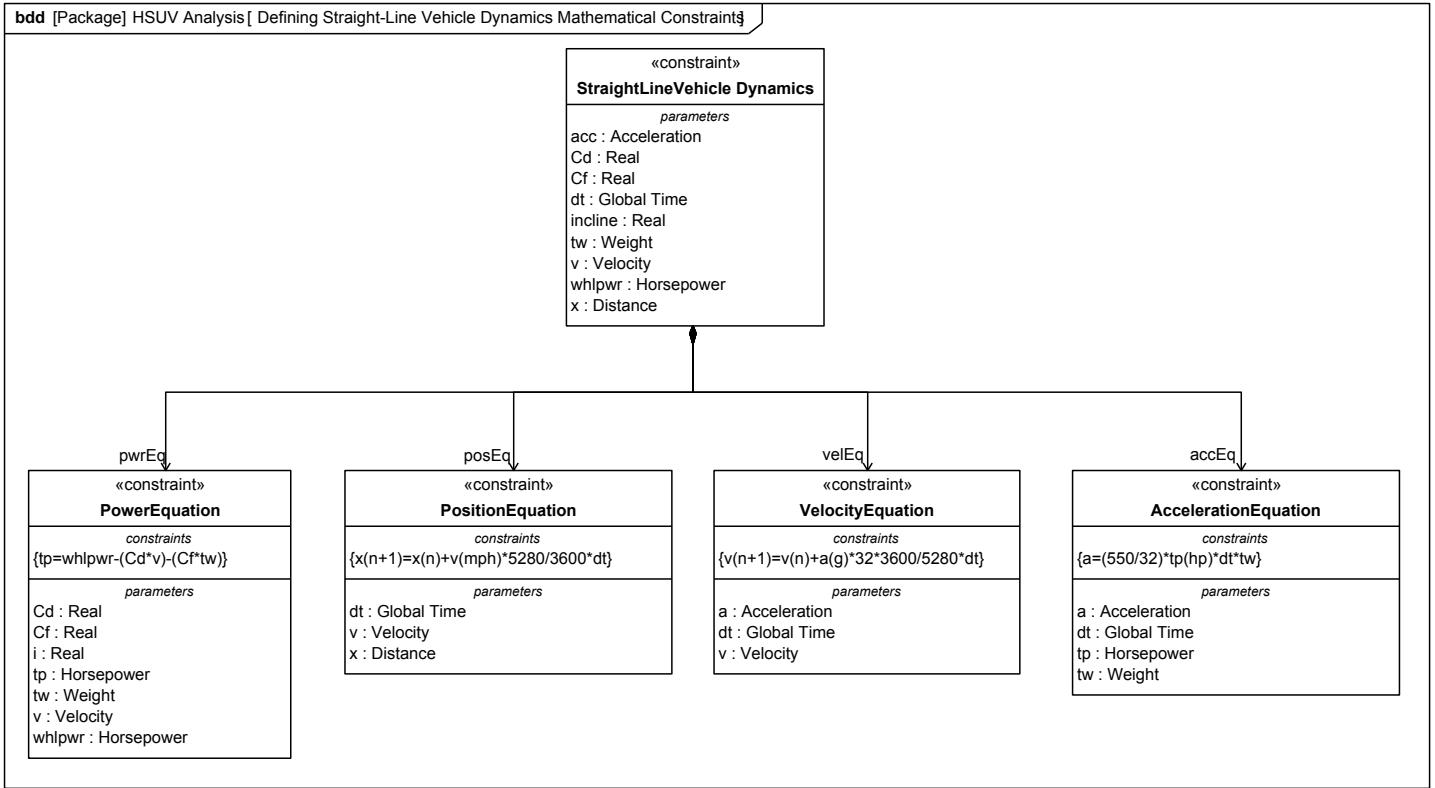


Figure D.34. Defining Straight-Line Vehicle Dynamics Mathematical Constraints

D.4.7.9 (Non-Normative) Non-SysML Diagram - 100hp Acceleration

Timing diagrams, while included in UML 2, are not directly supported by SysML. For illustration purposes, however, the interaction shown in [Fig. D.35](#) was generated based on the constraints and parameters of the `StraightLineVehicleDynamics` constraintBlock, as described in the [Fig. D.33](#). It assumes a constant 100hp at the drive wheels, 4000lb gross vehicle weight, and constant values for *Cd* and *Cf*.

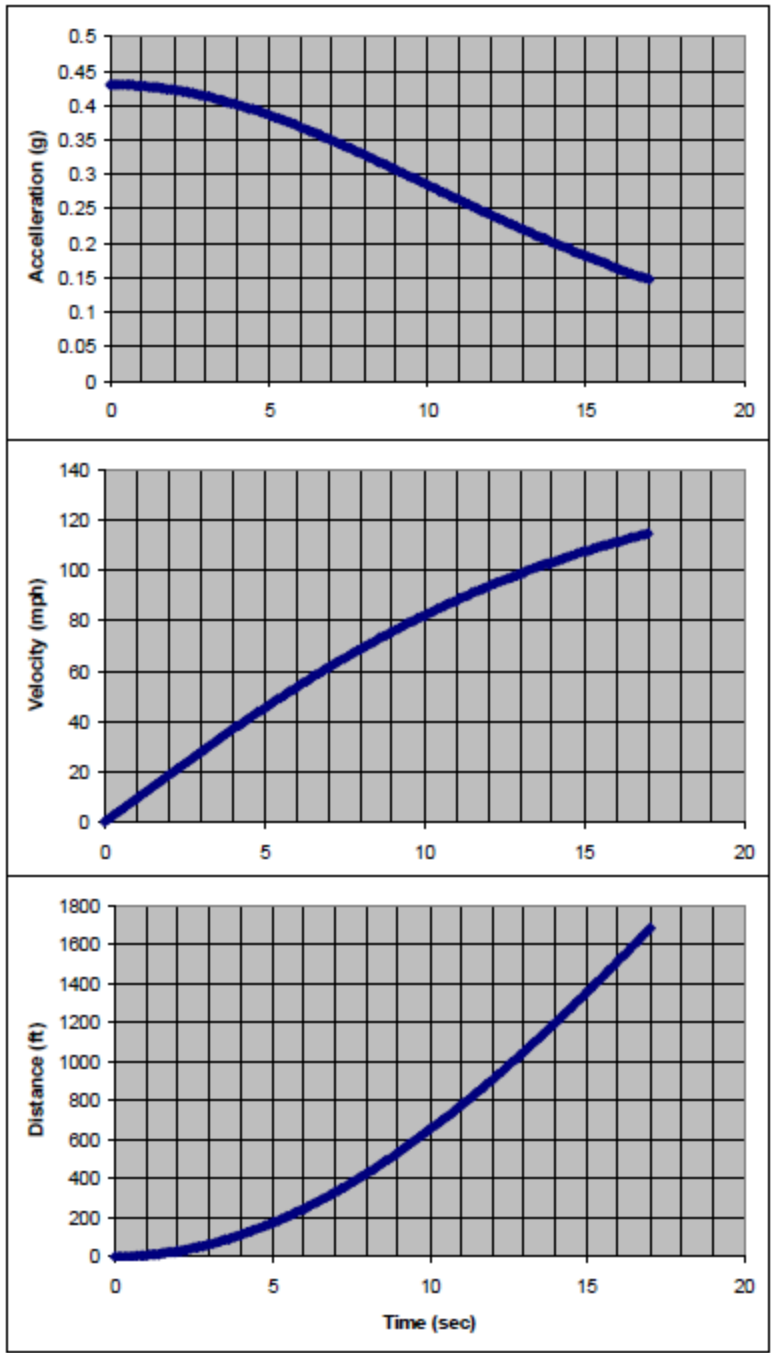


Figure D.35. 100 Wheel Horsepower

Please Note: This diagram are not a normative part of the SysML specification. So this diagram is for illustration and reference only.

D.4.8 Defining, Decomposing, and Allocating Activities

D.4.8.1 Activity Diagram - Acceleration (top level)

[Fig. D.36](#) shows the top level behavior of an activity representing acceleration of the HSUV. It is the intent of the systems engineer in this example to allocate this behavior to parts of the PowerSubsystem. It is quickly found, however, that the behavior as depicted cannot be allocated, and must be further decomposed.

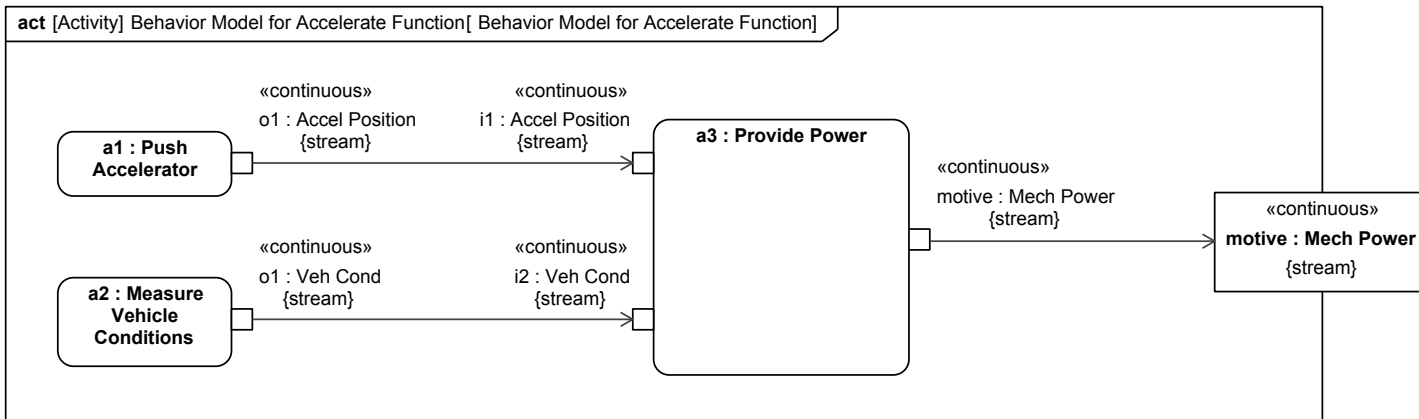


Figure D.36. Behavior Model for Accelerate Function

D.4.8.2 Block Definition Diagram - Acceleration

Fig. D.37 defines a decomposition of the activities from the activity diagram in Fig. D.36.

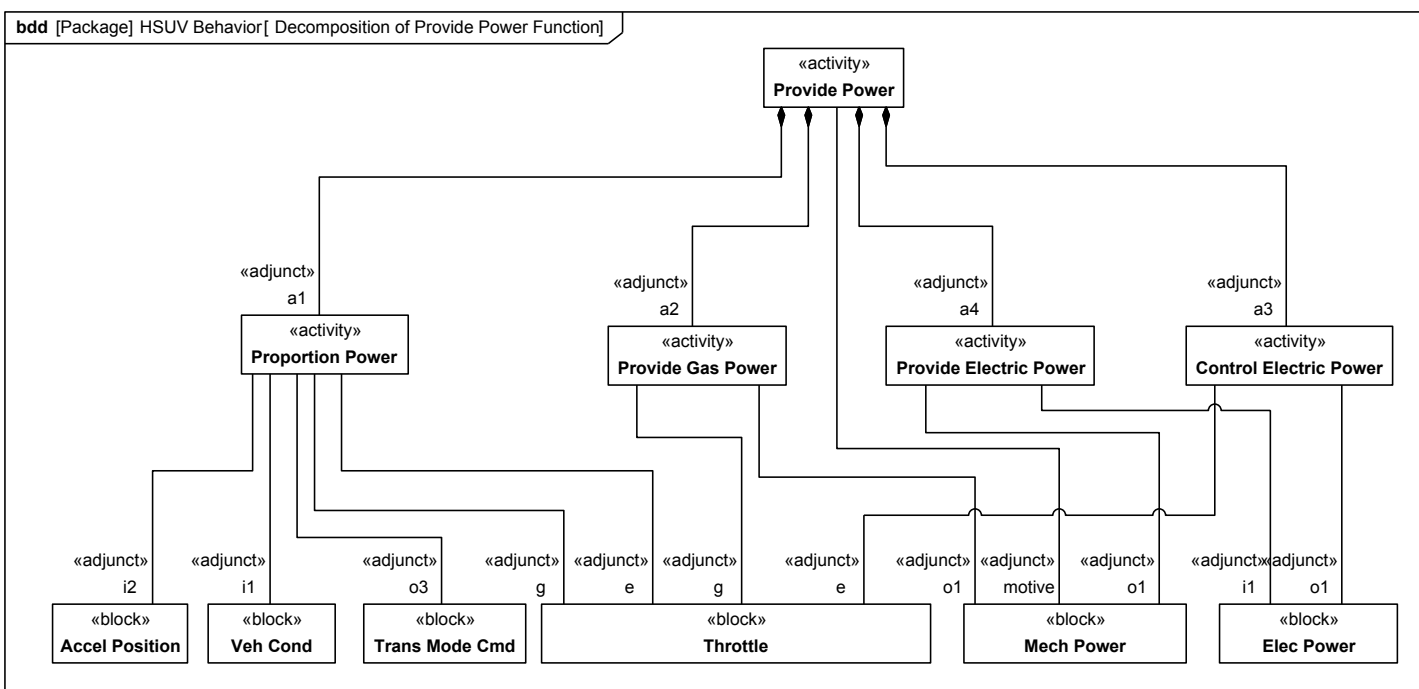


Figure D.37. Decomposition of Provide Power Function

D.4.8.3 Activity Diagram (EFFBD) - Acceleration (detail)

SysML allows the representation of relationships using tables without constraining the exact layout of such a table.

Section D.4.4.4 provides an example showing allocation relationships in tabular form. This is a more compact representation than the requirements diagrams shown previously.

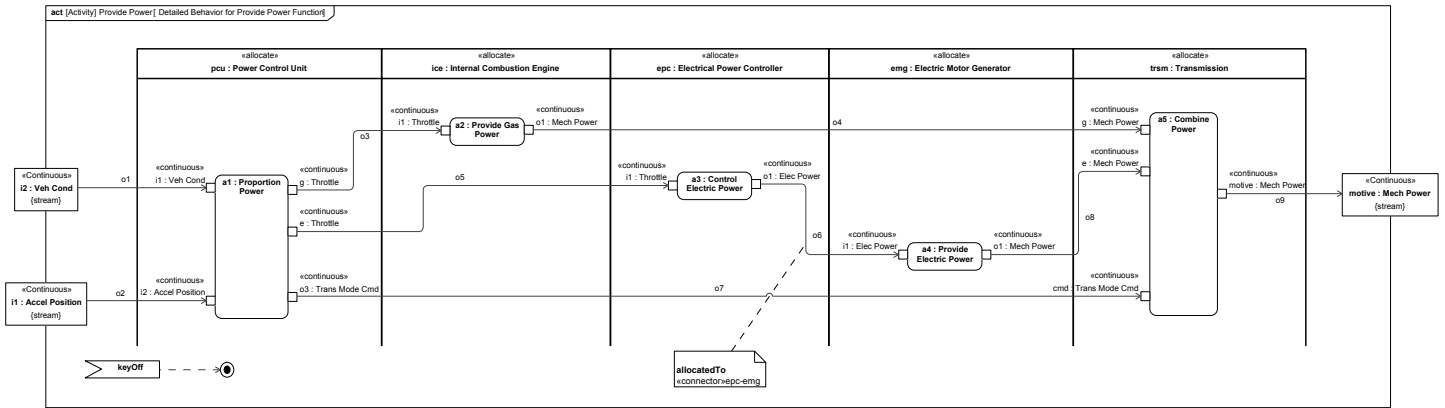


Figure D.38. Detailed Behavior for Provide Power Function

Note hierarchical consistency with [Fig. D.36](#)

D.4.8.4 Internal Block Diagram - Power Subsystem Behavioral and Flow Allocation

SysML allows the representation of relationships using tables without constraining the exact layout of such a table.

[Section D.4.4.4](#) provides two examples showing requirement containment (decomposition), and requirements derivation in tabular form. This is a more compact representation than the requirements diagrams shown previously.

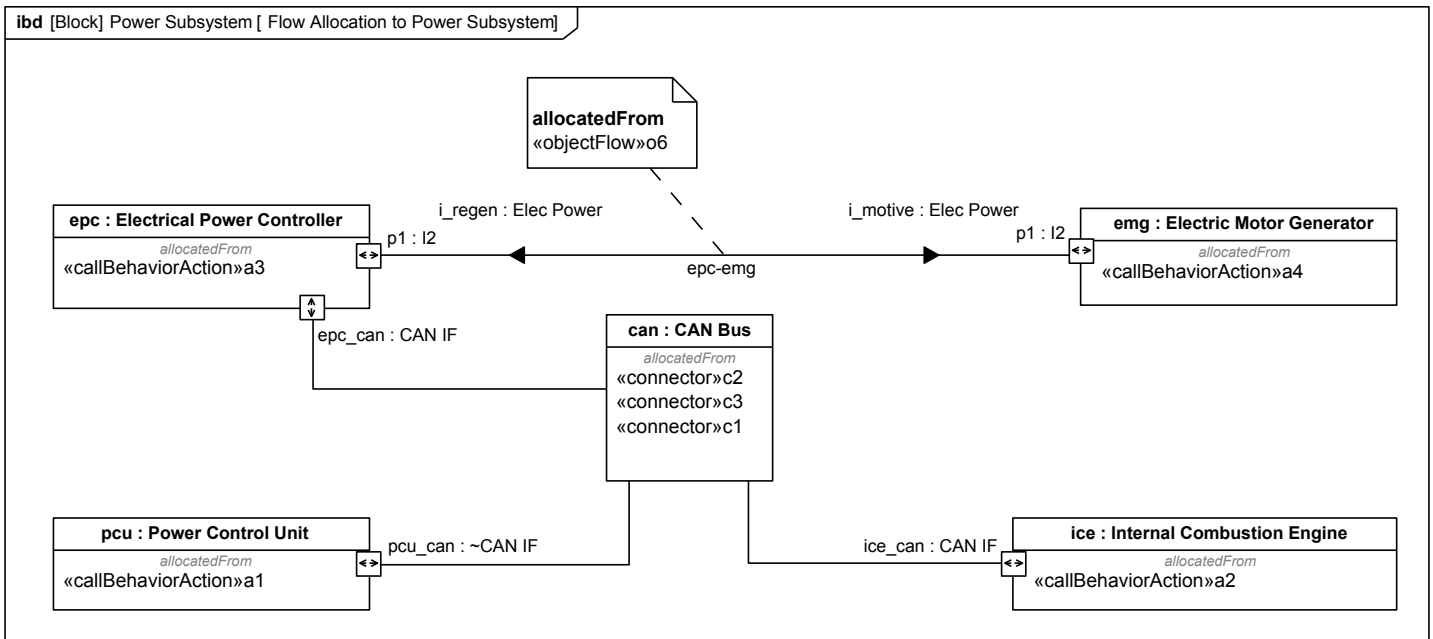


Figure D.39. Flow Allocation to Power Subsystem

D.4.8.5 Table - Acceleration Allocation

SysML allows the representation of relationships using tables without constraining the exact layout of such a table.

[Fig. D.40](#) is a simple table showing each end of the allocation relationships also shown in [Fig. D.38](#) and [Fig. D.39](#). This table is a more compact representation than the diagrams shown previously.

[Section D.4.4.4](#) also provided two examples showing requirement containment (decomposition), and requirements derivation in tabular form.













#	Applied Stereotype	Client	Supplier
1	<<> Allocate [Abstraction]	 a1:Proportion Power	 pcu : Power Control Unit
2	<<> Allocate [Abstraction]	 a2:Provide Gas Power	 ice : Internal Combustion Engine
3	<<> Allocate [Abstraction]	 a3:Control Electric Power	 epc : Electrical Power Controller
4	<<> Allocate [Abstraction]	 a4:Provide Electric Power	 emg : Electric Motor Generator
5	<<> Allocate [Abstraction]	 a5:Combine Power	 trsm : Transmission
6	<<> Allocate [Abstraction]	 Object Flow:o6[o1 -> i1]	 Connector:epc-emg[epc.p1 - emg.p1]

Figure D.40. Tabular Representation of Allocation from “Accelerate” Behavior Model to Power Subsystem (Table)

Please Note: Tables are not a normative part of the SysML specification. So all tables are for illustration and reference only.

D.4.8.6 Block Definition Diagram: Slot Values - EPA Fuel Economy Test

[Fig. D.41](#) demonstrates the use of InstanceSpecifications to show a particular Hybrid SUV (VIN number provided as a slot value) satisfying the EPA fuel economy test. Serial numbers of specific relevant parts are also indicated as slot values.

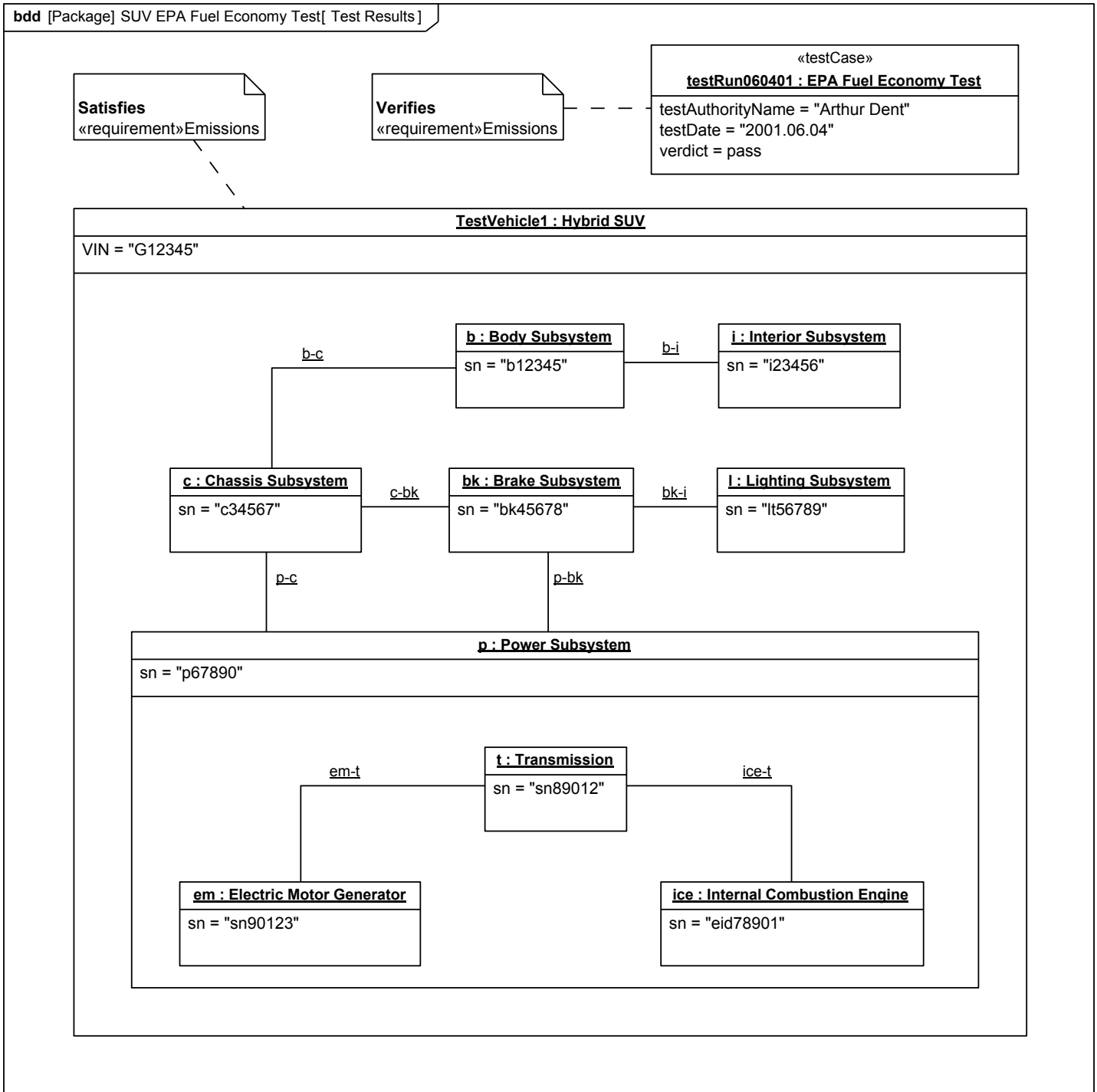


Figure D.41. Test Results