Papyrus Tutorial

A Practical Guide to Systems Modeling
with Papyrus and SysML

Version 0.1.1

Contents

List of Figures

1 INTRODUCTION

1.1 Overview

The purpose of this tutorial is to use the *Papyrus* program to create a model of a simple system utilizing the SysML language. A vehicle climate control unit will be modeled starting with the Problem Domain and then moving to the Solution Domain.

1.2 Assumptions

This tutorial assumes that *Papyrus* and the SysML tools have been properly installed and will not cover the procedures to do so. This is not a SysML tutorial but rather how to model an example system with SysML in the Papyrus program. This tutorial uses *Papyrus* version 2019-03 and SysML 1.4 running on a Windows machine.

2 PROBLEM DOMAIN

2.1 Introduction

In the problem domain we will analyze stakeholder needs and refine them with SysML model elements to gain a clear understanding of the problem that the System Of Interest (SoI) must solve. We will first analyze the SoI from a black-box perspective where we focus on the interaction of the system with the environment without knowledge of the internal structure of the SoI. We then analyze the system from a white-box perspective to perform the functional analysis of the system.

2.2 Black-box perspective

We will start this project by organizing our model and capturing our stakeholder needs.

2.2.1 Step 1. Create the new project and new model.

1. Create a new *Papyrus* project by selecting **File > New > Papyrus Project** (see figure 1).

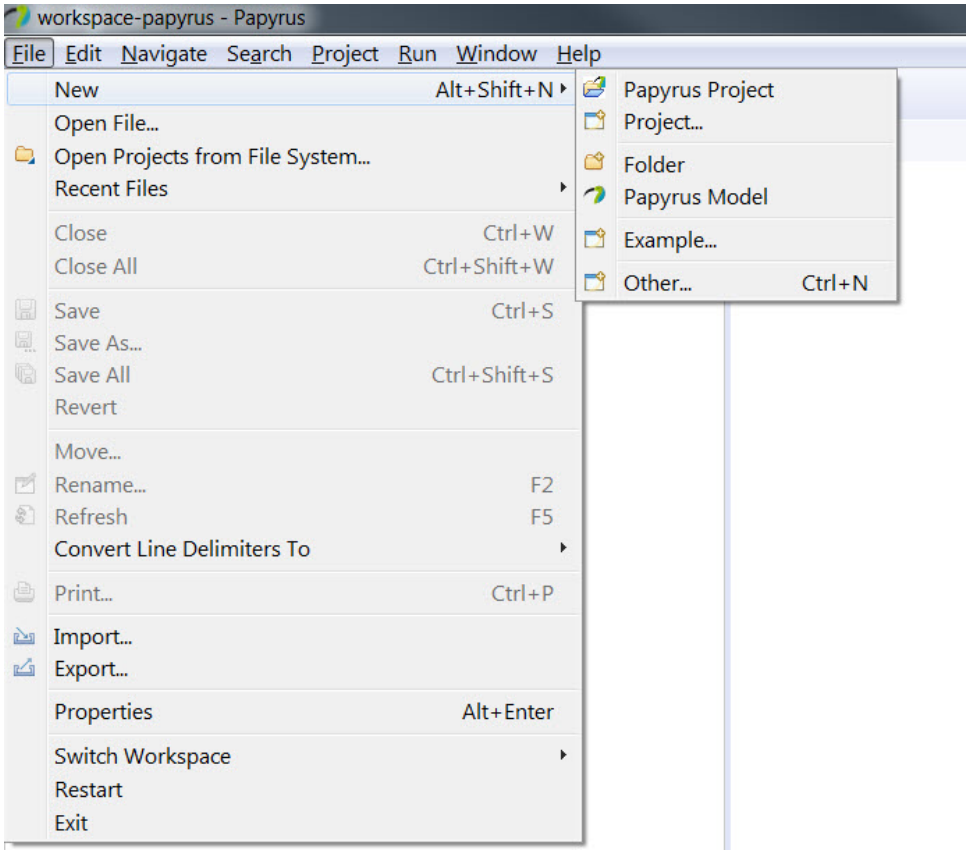

Figure 1: New Papyrus Project

2. The *Select Architecture Context* dialog will then appear Ensure that the *SysML 1.4* checkbox is selected and then select **Next** (see figure 2).

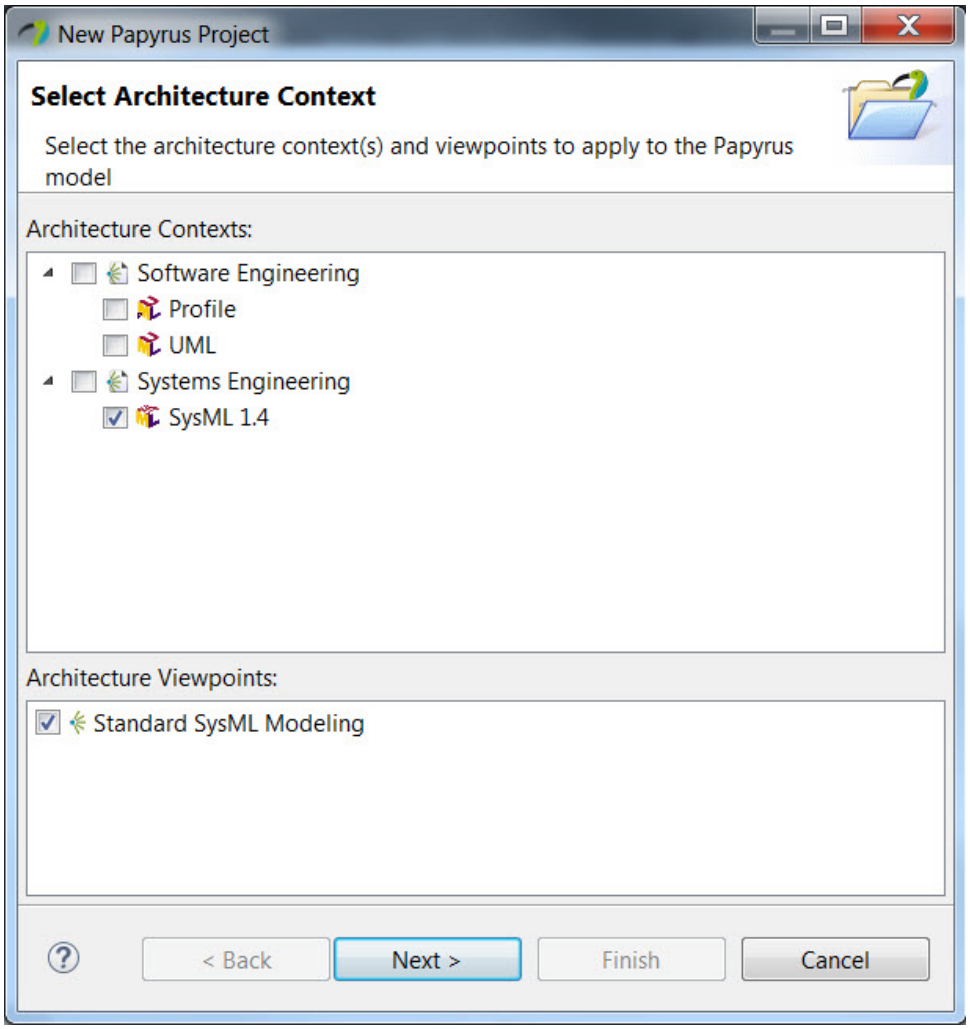

Figure 2: Select Architecture Context

3. On the next screen type *car* for the project name and click **Finish**. Your screen should look like figure 3.

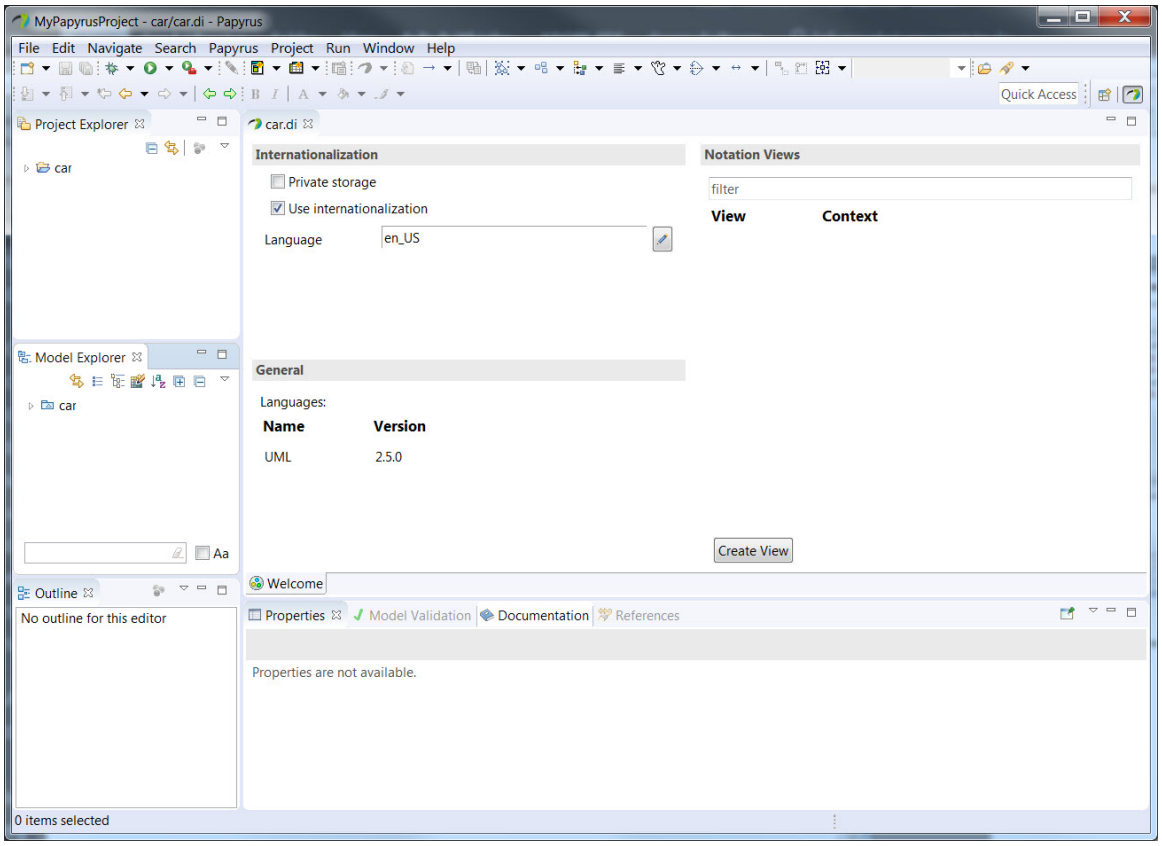

Figure 3: Project File Created

4. Now we will create our model for the vehicle climate control system problem space. Create a new *Papyrus* model by selecting **File > New > Papyrus Model** (see figure 4). Ensure that the *SysML 1.4* checkbox is selected when the *Select Architecture Context* dialog appears and then select **Next** (see figure 2).

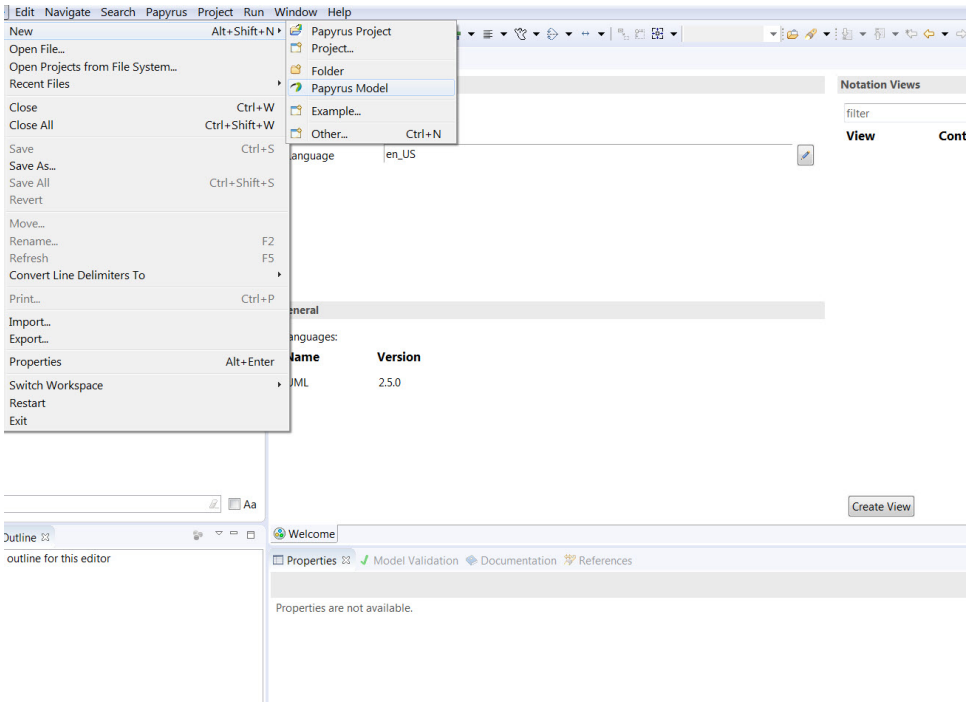

Figure 4: Create New Model

5. When the *New Papyrus Model* dialog appears, click on the *car* folder to choose this as the parent folder and enter *VehicleCCU_Problem* in the file name box and click **Finish** (see figure 5).

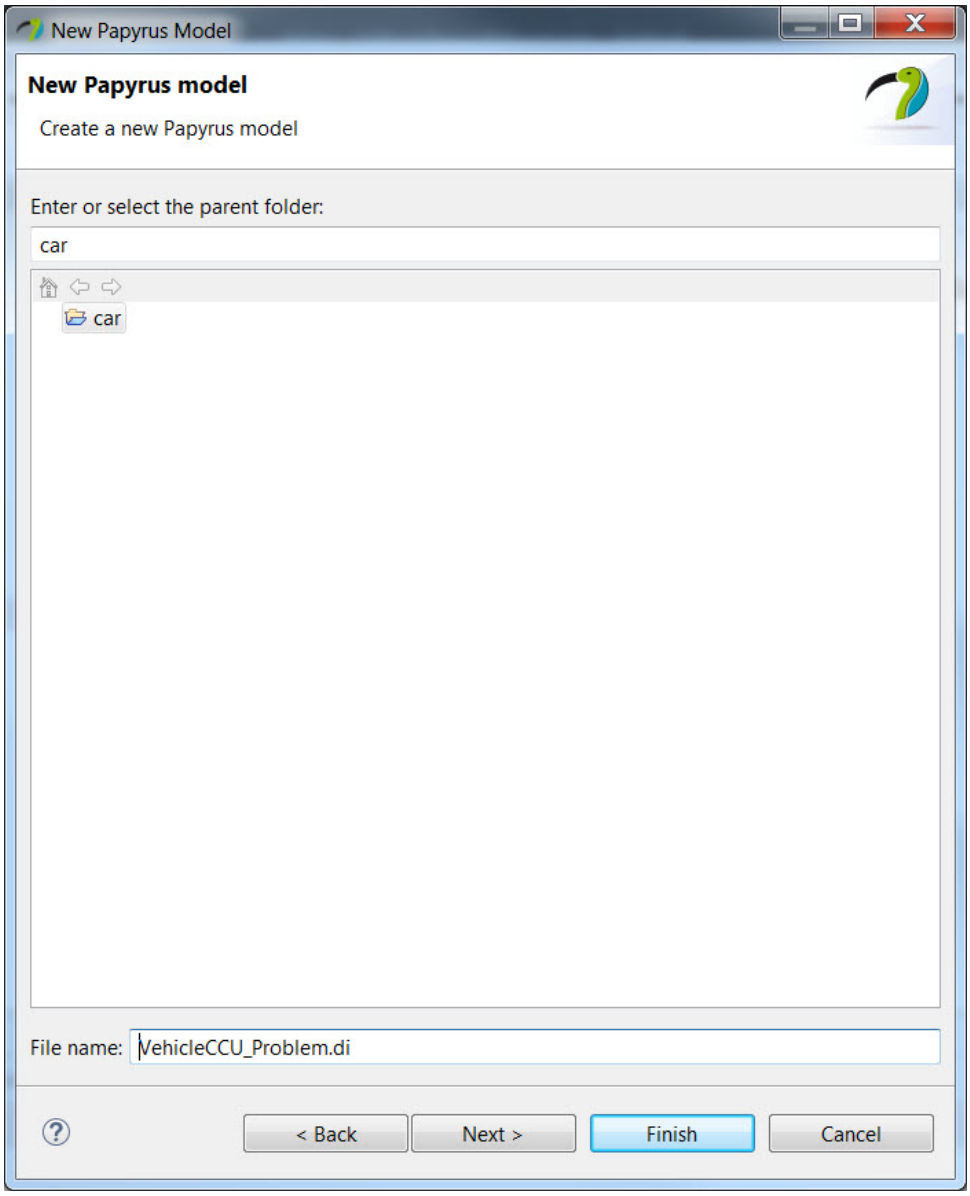

Figure 5: New Model Dialog

6. Your screen should now look like figure 6. You see that in the *Project Explorer* window we have the *car* folder with a car model and the model we created called *VehicleCCU_Problem*. The *Model Explorer* window should be showing the top level folder of our *VehicleCCU_Problem* model that we just created (figure 7). If it does not then double-click on the *VehicleCCU_Problem* in the *Project Explorer* window OR select the *VehicleCCU_Problem* tab in the main window.

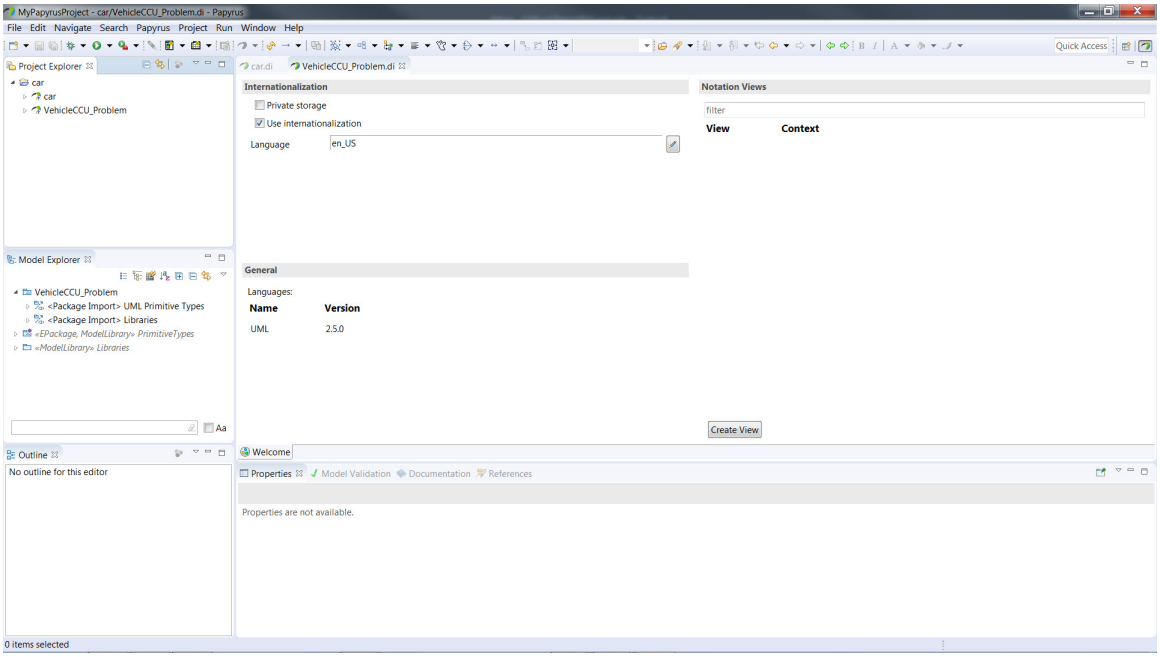

Figure 6: Vehicle CCU Problem Model

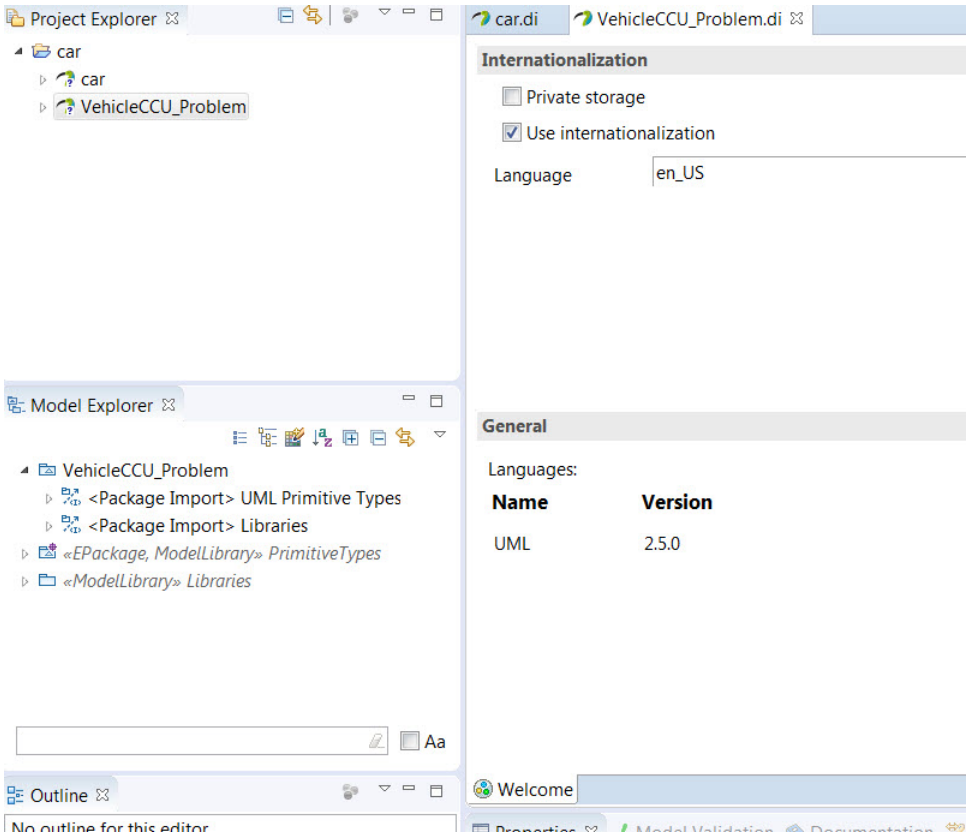

Figure 7: Vehicle CCU in Model Explorer

2.2.2 Step 2. Organize the model for Black-Box analysis.

1. We are now going to organize our model by creating three new packages in our *VehicleCCU_Problem* model and when we are finished it should look like figure 8.

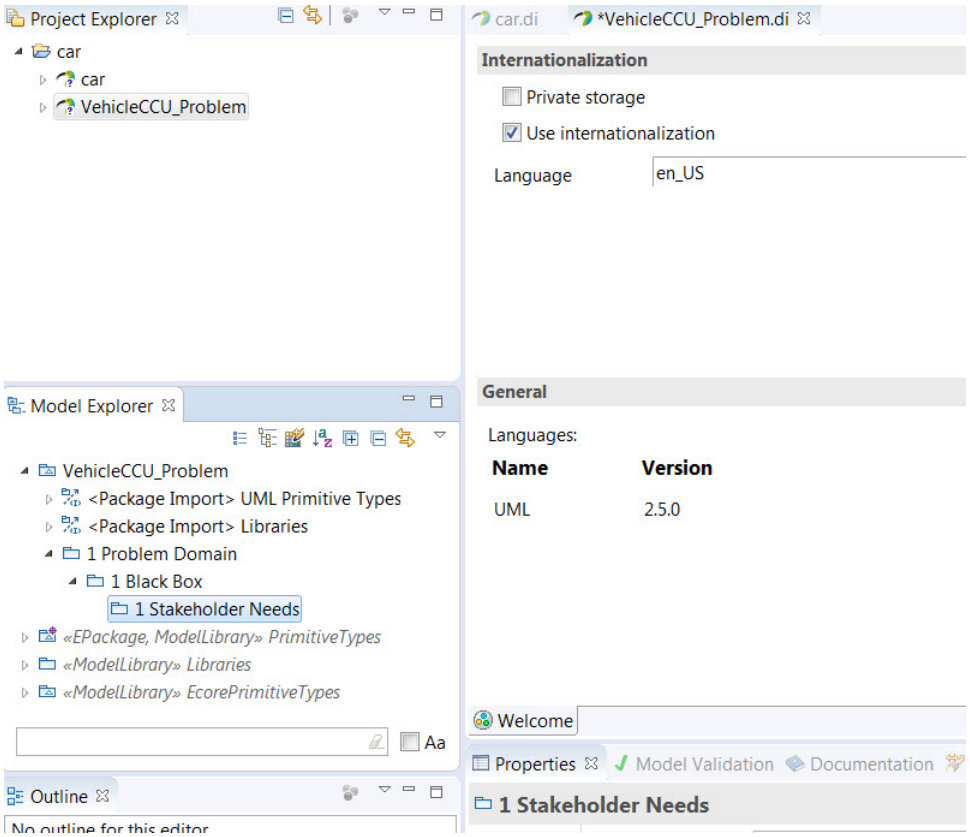

Figure 8: Black Box Packages

2. Right-click on the *VehicleCCU_Problem* top level folder in the *Model Explorer* window and then select **New Child** and then **Package** (figure 9). The new package will appear under the *VehicleCCU_Problem* folder. Name the package *1 Problem Domain*. (Note that if you have already hit enter the package is named with a default name like *Package14* and it can be renamed in the UML properties tab at the bottom of the main window.

3. Now right-click on the *1 Problem Domain* package and follow the same procedure to create a new child package named *1 Black Box*.

4. Now right-click on the *1 Black Box* package and follow the same procedure to create a new child package named *1 Stakeholder Needs*. Your result should look like figure 8. If you make a mistake along the way you can simply delete a package and try again. Ensure that the packages are nested as shown in figure 8 before continuing.

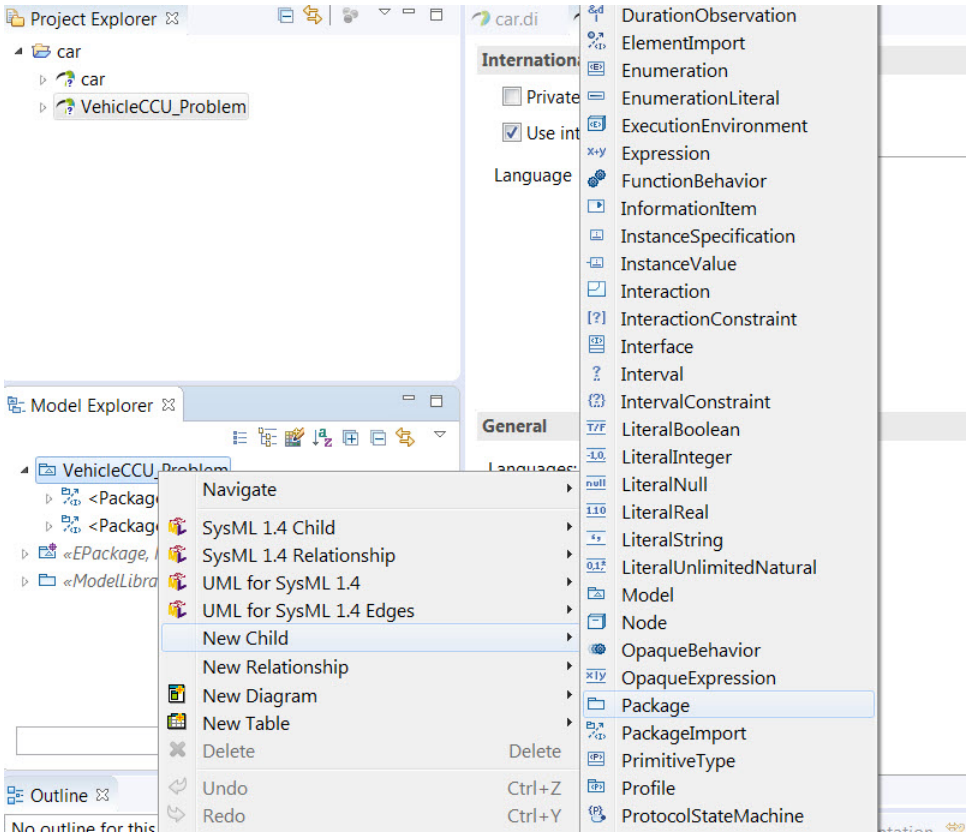

Figure 9: Create a New Package

2.2.3 Step 3. Capturing Stakeholder Needs (Requirements Table)

1. After talking with our stakeholders we will capture our initial list of stakeholder needs for the Vehicle Climate Control Unit using a SysML Requirement Table. There would of course be many requirements but we will just capture a few for illustrative purposes.

2. Right-click on the *1 Stakeholder Needs* package in the *Model Explorer* window and select **New Table > SysML 1.4 Requirement Tree Table** (figure 10). Name the table *Stakeholder Needs*. We want to use the Requirement Tree Table so we can have nested requirements in our requirements table.

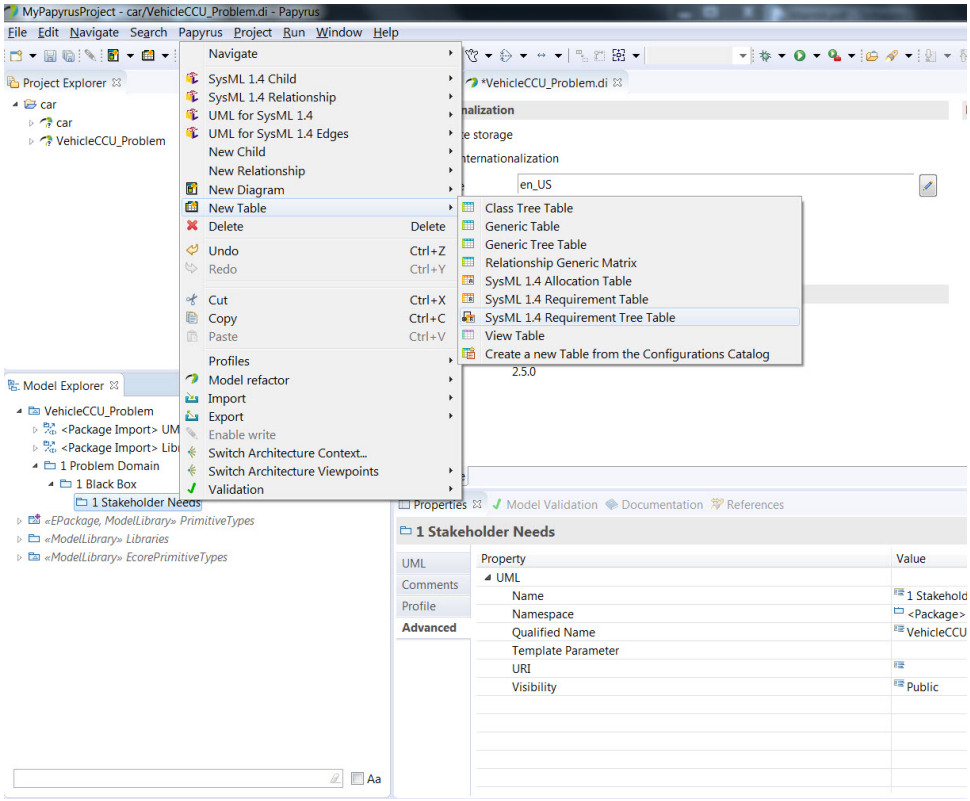

Figure 10: Creating a New Requirements Table

3. You should now see an initial table in the main window with two cells at the top labeled *id* and *text*.

4. First Left-click anywhere in the blank space in the main table window and then Right-click and select **Create SysML 1.4 Element > Create New Requirement** (see figure 11). A new requirement should appear in the table view and a requirement model element should appear underneath the *Stakeholder Needs* table element in the *Model Explorer* window (figure 12).

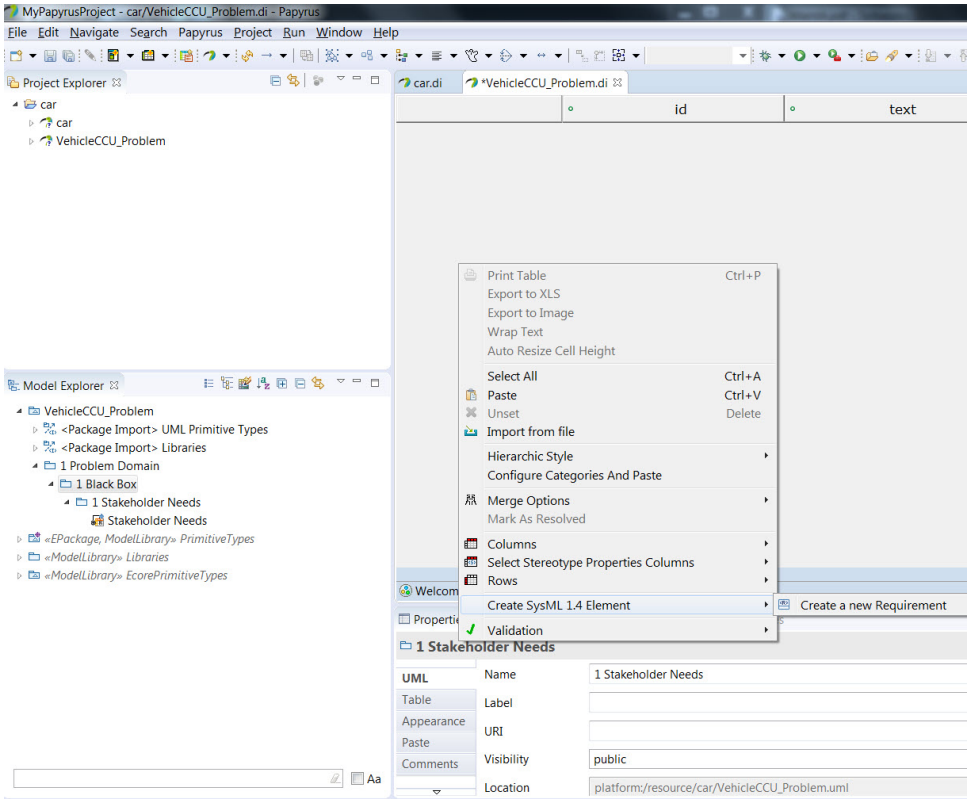

Figure 11: Create New Requirement in Requirement Table

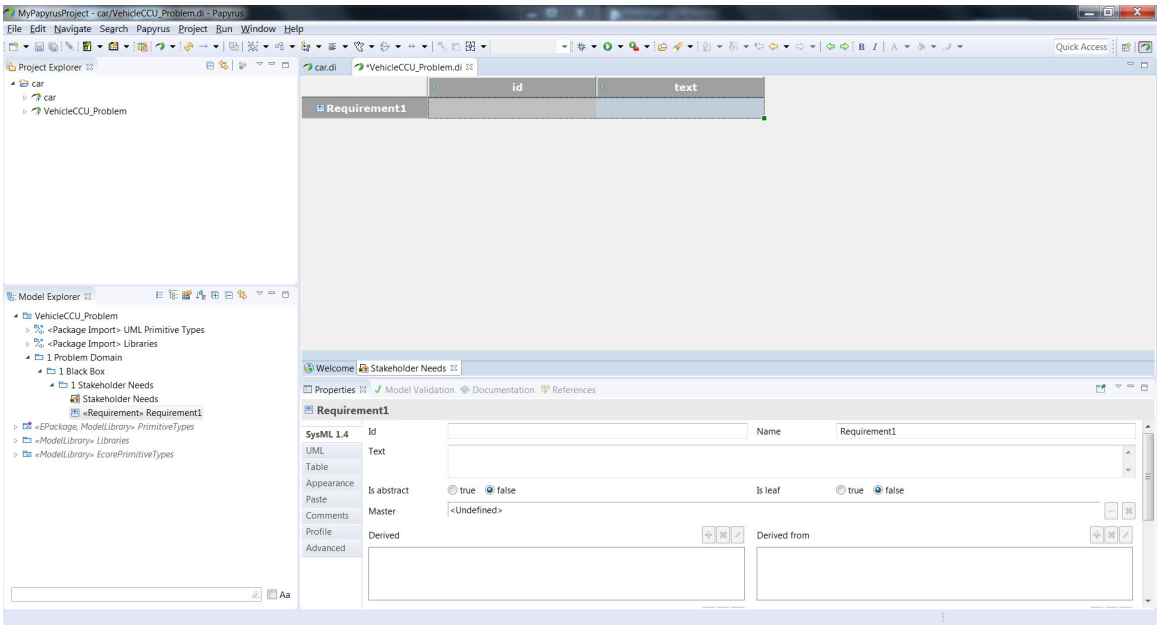

Figure 12: New Requirement in Model

5. Most software like *Papyrus* has multiple ways to accomplish the same task. We can type directly in the requirements table cells to enter the *id* and *text* or we can enter it

in the *Properties Window* in the *SysML 1.4* tab. We are going to use the *SysML 1.4* tab at the bottom of the main screen . To do this Left-click on *Requirement 1* under the *Stakeholder Needs* table element in the *Model Explorer* then select the *SysML 1.4* tab in the *Properties* window (figure 12). You can see the properties in this tab for *Id*, *Name*, and *Text*.

6. Type *SN-1* in the *Id* field and *SN-1 User Needs* in the *Name* field. We will not enter anything in the *Text* field as this will be a heading for our user needs (figure 13).

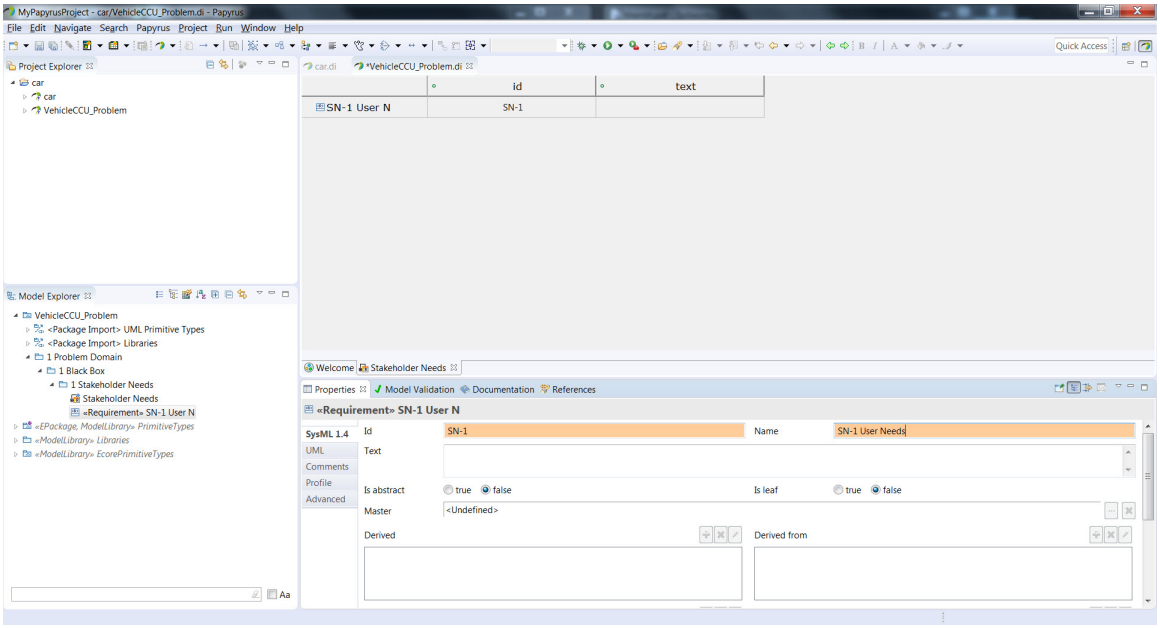

Figure 13: User Needs

7. Continue adding requirements and populating the *Id*, *Name*, and *Text* fields until the requirements table looks like figure 14.

	id	text
SN-1 User Needs	SN-1	
SN-1.1 Setting Temperature	SN-1.1	It must be possible to set and maintain desired temperature in the car.
SN-1.2 Heat and Cool Modes	SN-1.2	Unit shall heat and cool.
SN-1.3 Noise Level	SN-1.3	Climate control unit in max mode shall not be louder than the engine.
SN-1.4 Climate Control Mass	SN-1.4	Mass of the unit shall not exceed 1 percent of the total car mass.

Figure 14: User Needs Requirements Table

8. In the *Model View* window Left-click on the requirement *SN-1.1 Setting Temperature* requirement then hold down the Control key and select the last requirement *SN-1.4 Climate Control Mass*. Drag these requirements and drop them on the *SN-1 User Needs* to make these requirements child requirements of our heading requirement. Your screen should look like figure 15 with the requirements nested under *SN-1 User Needs* in the *Model View* window and a tree structure in the table view.

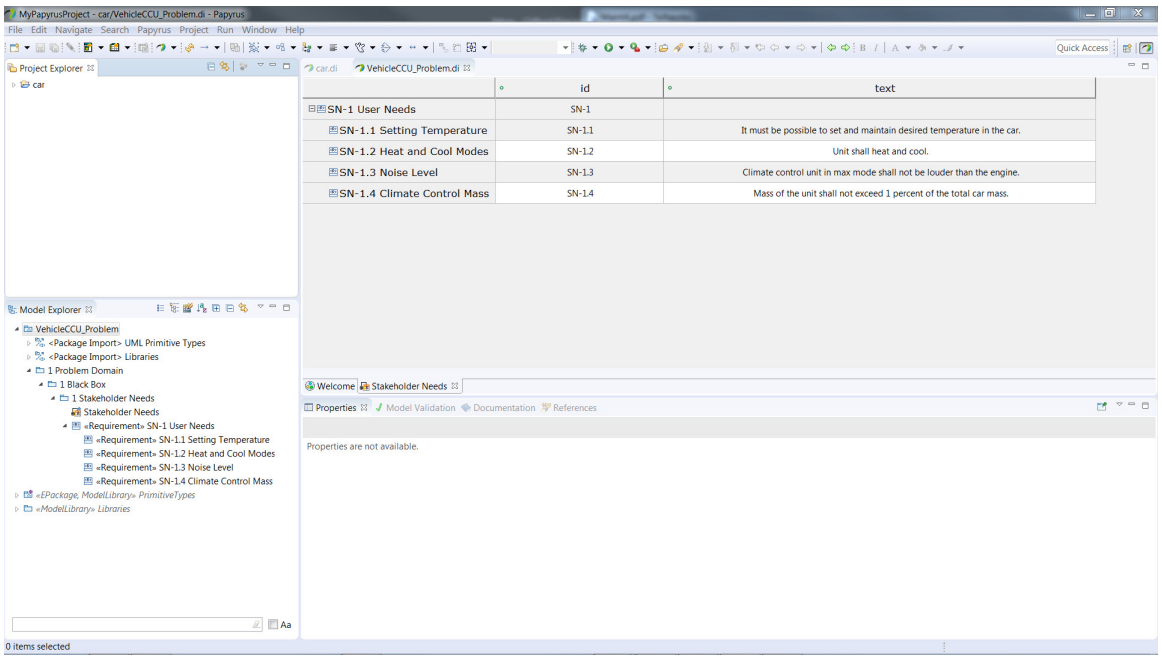

Figure 15: Requirements Tree

2.2.4 Step 4. Initial System Context

The operating environment or System Context defines the external view of the system where the SoI is treated as a black box. It introduces elements that interact with the system but are not owned by the system. During this initial phase of system context definition one or more system contexts are defined and elements that participate in each context are identified. Several sources recommend using blocks rather than actors to define humans for the following reasons:

- A block is treated as a part of the system context. An actor is not.
- A block can be marked as external. An actor cannot.
- A block can have its behavior defined. An actor cannot.

Let's start modeling the Initial System Context.

1. Set up a new package under the *1 Black Box* package named *3 System Context* by Right-clicking on the *1 Black Box* package in the *Model Explorer* window and selecting **New Child** and then **Package**. Name the new package *3 System Context* (figure 16).

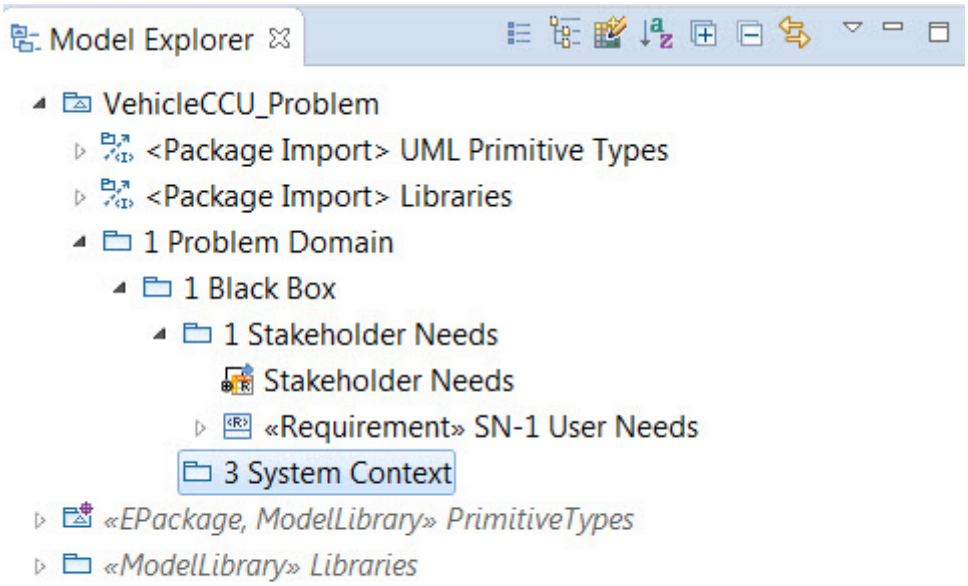

Figure 16: System Context package

2. Now we will create a block that captures the vehicle in use system context. Right-click on the *3 System Context* package just created and select **SysML 1.4 Child** and then **Block** (figure 17). Name the block *Vehicle In Use*. Your model explorer should now look like figure 18.

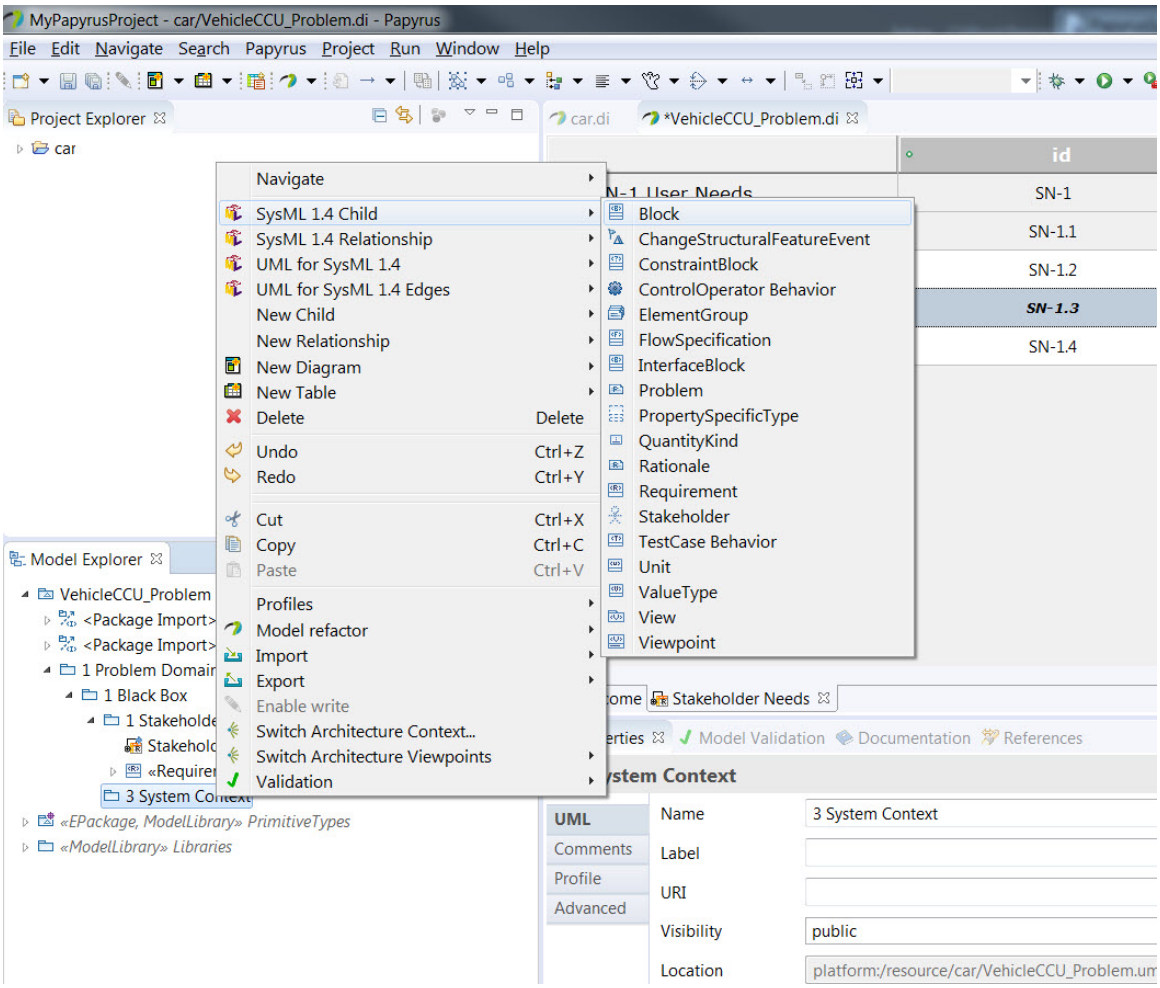

Figure 17: Create New Block

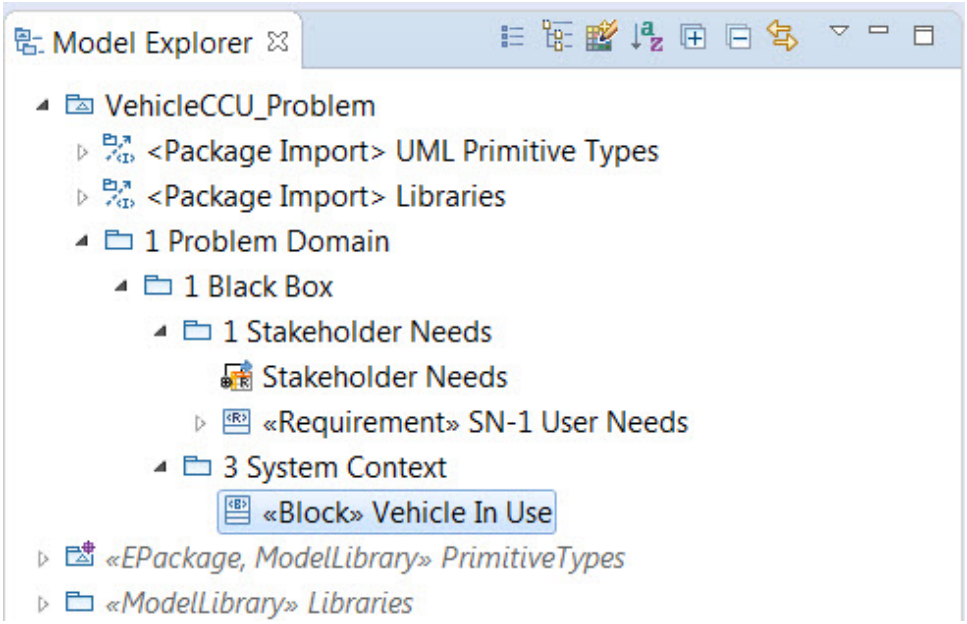

Figure 18: Vehicle In Use Block

3. To specify participants of the Vehicle In Use context we will create a SysML Internal Block Diagram (IBD). Right-click on the *Vehicle In Use* block just created and select **New Diagram** then **SysML 1.4 Internal Block Diagram** (figure 19). Name the new diagram *Vehicle In Use*, the same as the block. You should see the diagram in the main window (figure 20). If you do not, simply Double-click on the *Vehicle In Use* internal block diagram in the *Model Explorer* window.

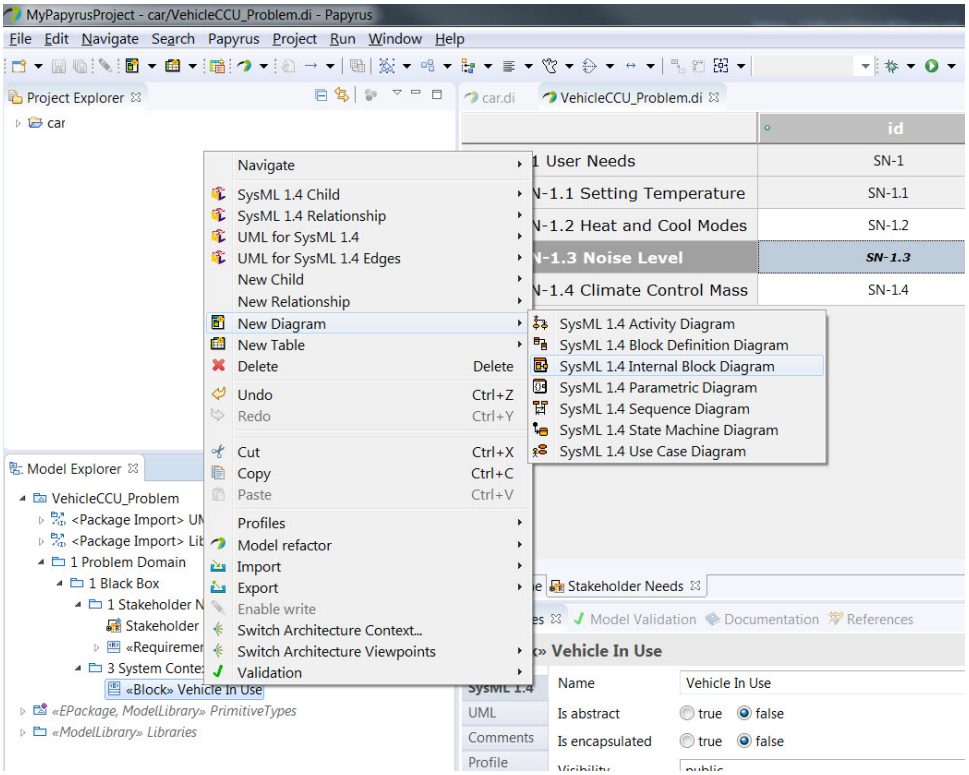

Figure 19: Create Internal Block Diagram

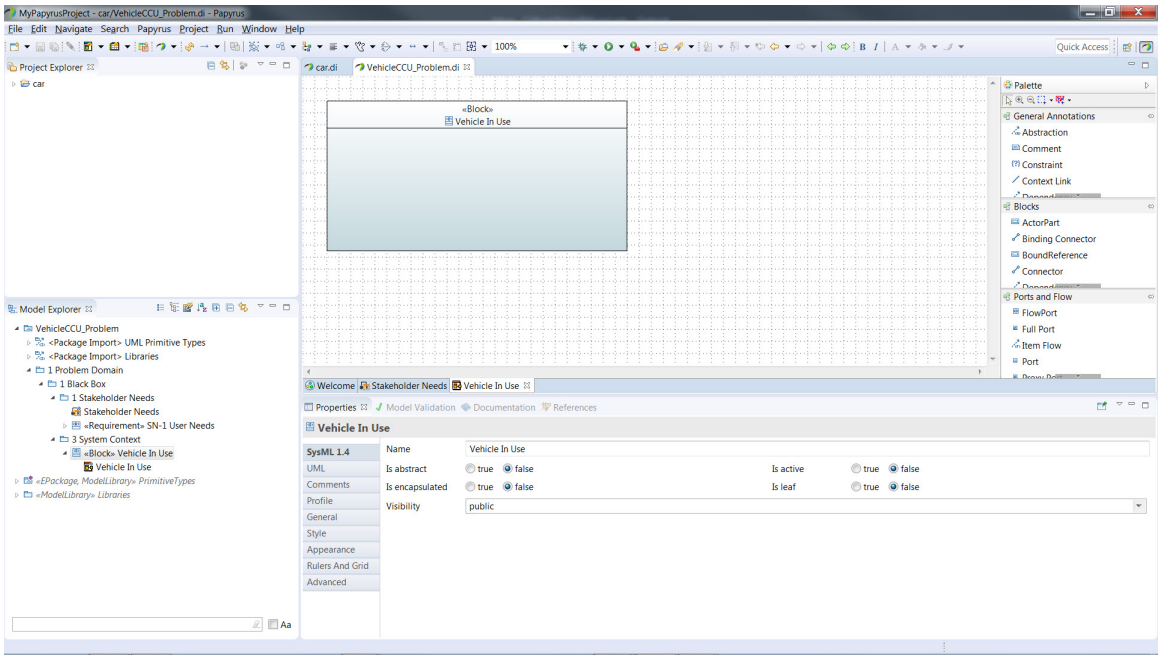

Figure 20: Vehicle In Use Internal Block Diagram

4. Our analysis of stakeholder needs indicates that the Vehicle In Use system context includes the following participants: the vehicle Climate Control Unit, Vehicle Occu-

pant, and an Energy Supply of some kind to power everything. The participants can be captured as part properties in the IBD. Open the *Vehicle In Use* IBD you just created if not already open. Use the tool palette (If the tool palette is not visible click on the small arrow on the right side of the main window vertical scroll bar to display it) on the right and select **Part** from the **Blocks** diagram palette . Now Left-click anywhere inside the *Vehicle In Use* diagram in the main window and the *Type Initialization* window will appear (figure 21).

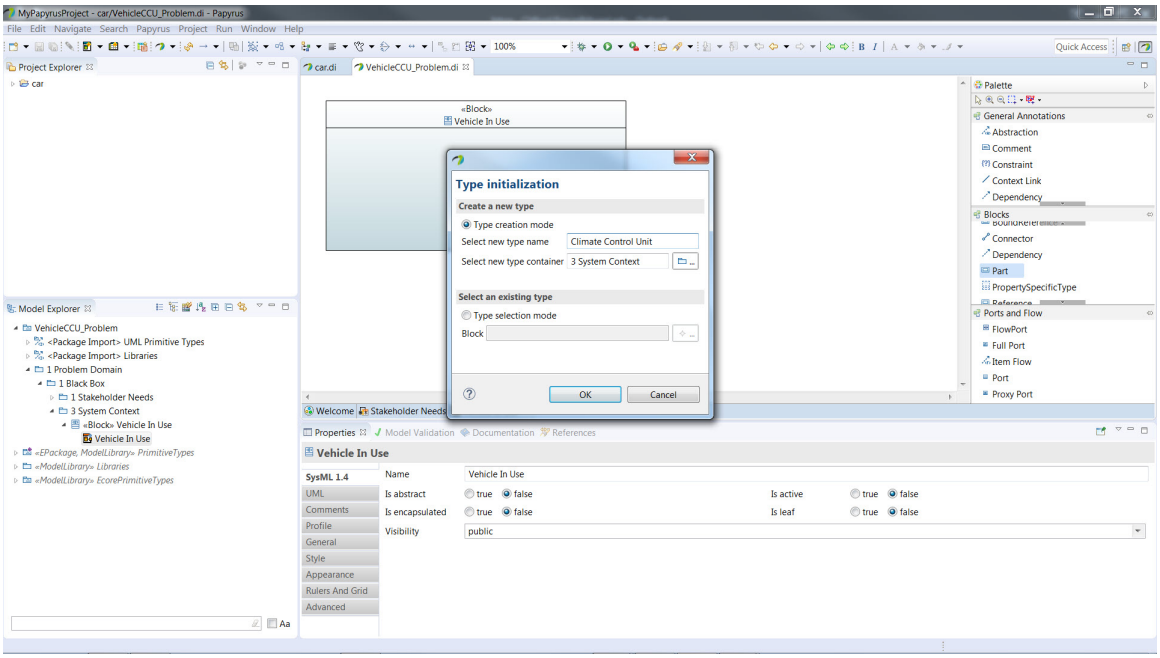

Figure 21: Part Type Initialization Window

5. In the *Type Initialization* window ensure that the *Type creation mode* radio button is selected to create a new type. In the *Select new type name box* type *Climate Control Unit* and press *Enter* twice. Your model should look like figure 22.

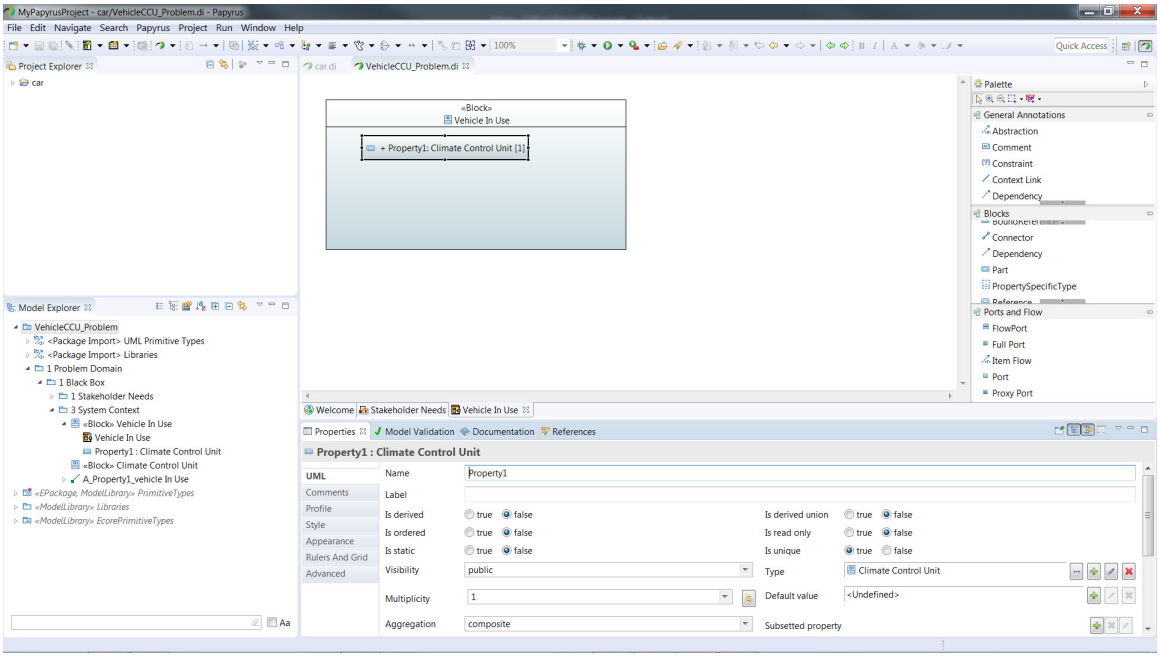

Figure 22: Part Type Initialization Window

6. Let's make this new block a little cleaner by not displaying the *Property1* name in front of the colon in the *Climate Control Unit* part in the diagram. Select the *Climate Control Unit* in the diagram. In the *Properties* window at the bottom of the screen select the *UML* tab and place your cursor in the *Label* field. Type a single *Space* with the spacebar and hit *Enter*. Your *Climate Control Unit* should now look like figure 23.

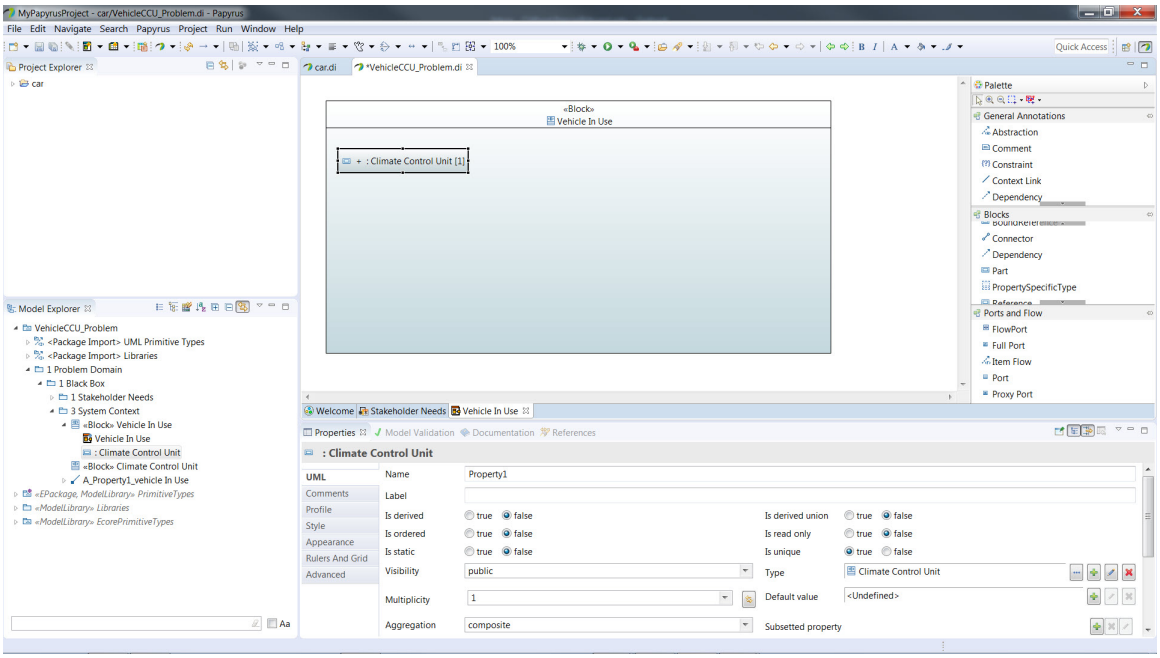

Figure 23: Climate Control Unit with Label Removed

7. Create two more parts; one called *Vehicle Occupant* and one called *Energy Supply*. When finished your diagram should look like figure 24.

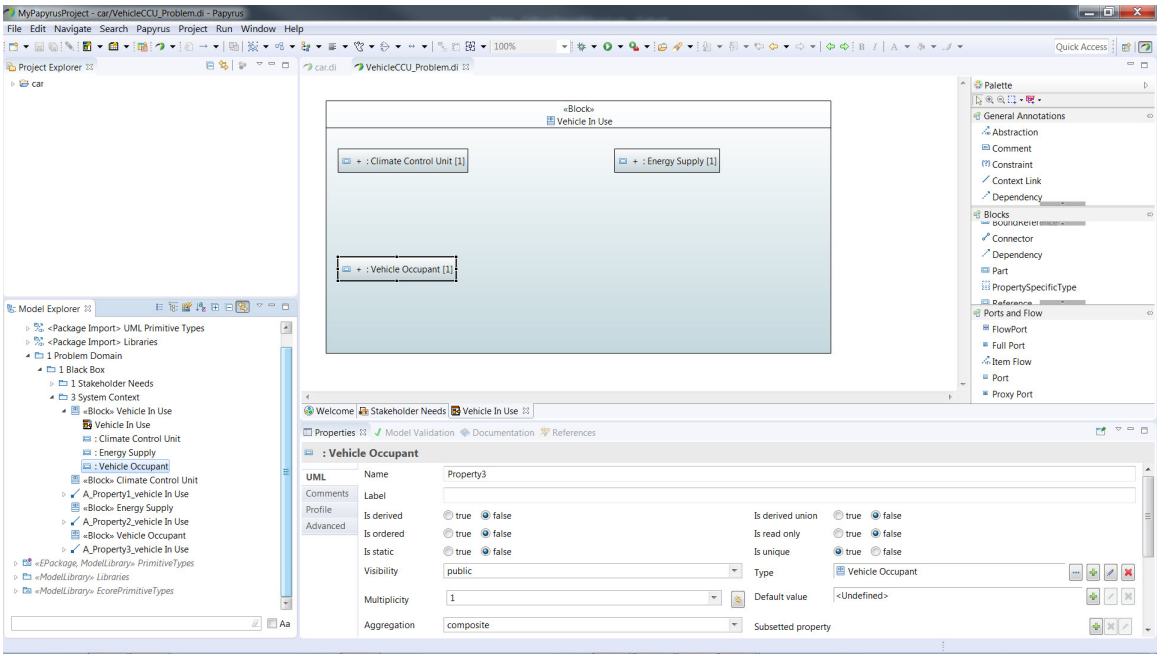

Figure 24: Vehicle In Use IBD

2.2.5 Step 5. Use Cases

In this section, the functional stakeholder needs will be refined with use cases and use case scenarios. Again we will be using blocks rather than actors (see section 2.2.4).

1. Set up a new package under the *1 Black Box* package named *2 Use Cases* by Right-clicking on the *1 Black Box* package in the *Model Explorer* window and selecting **New Child** and then **Package**. Name the new package *2 Use Cases* (figure 25).

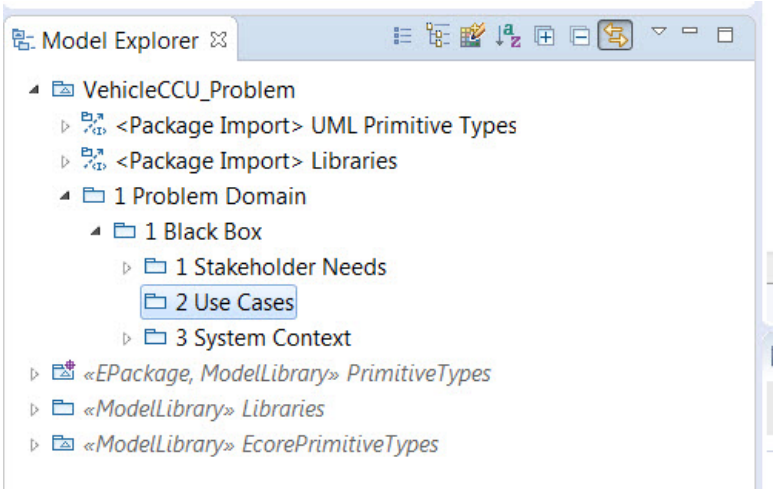

Figure 25: Use Case Package

2. Now that we have created the use case package we need to create a Use Case Diagram. In the *Model Explorer* window, Right-click on the *2 Use Cases* package that you created in the previous step and select *New Diagram* and then *SysML 1.4 Use Case Diagram* (figure 26). Name the diagram *Use Cases of Vehicle In Use SC*. The new use case diagram will appear under your *2 Use Cases* package and a new blank use case diagram should appear in the main window (figure 27).

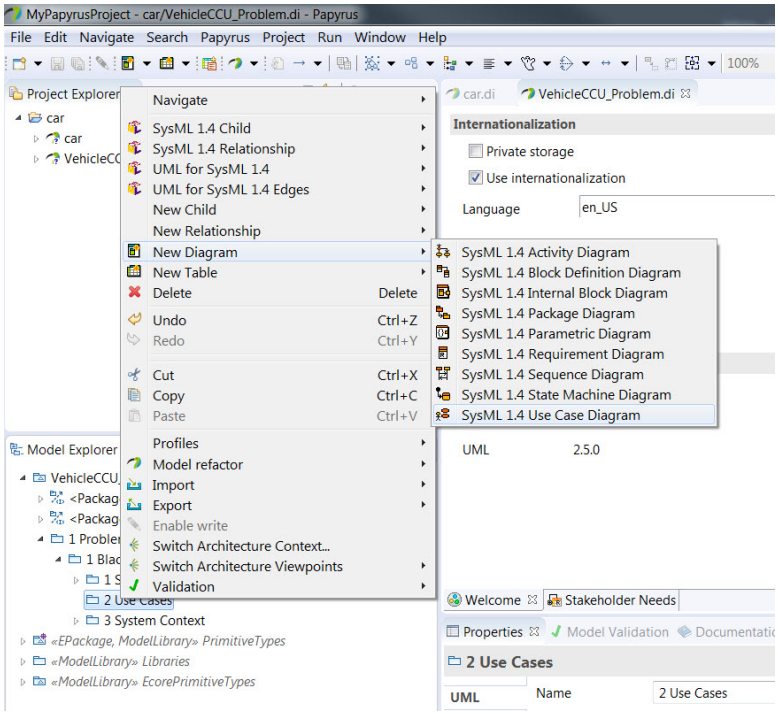

Figure 26: Use Case Diagram Creation

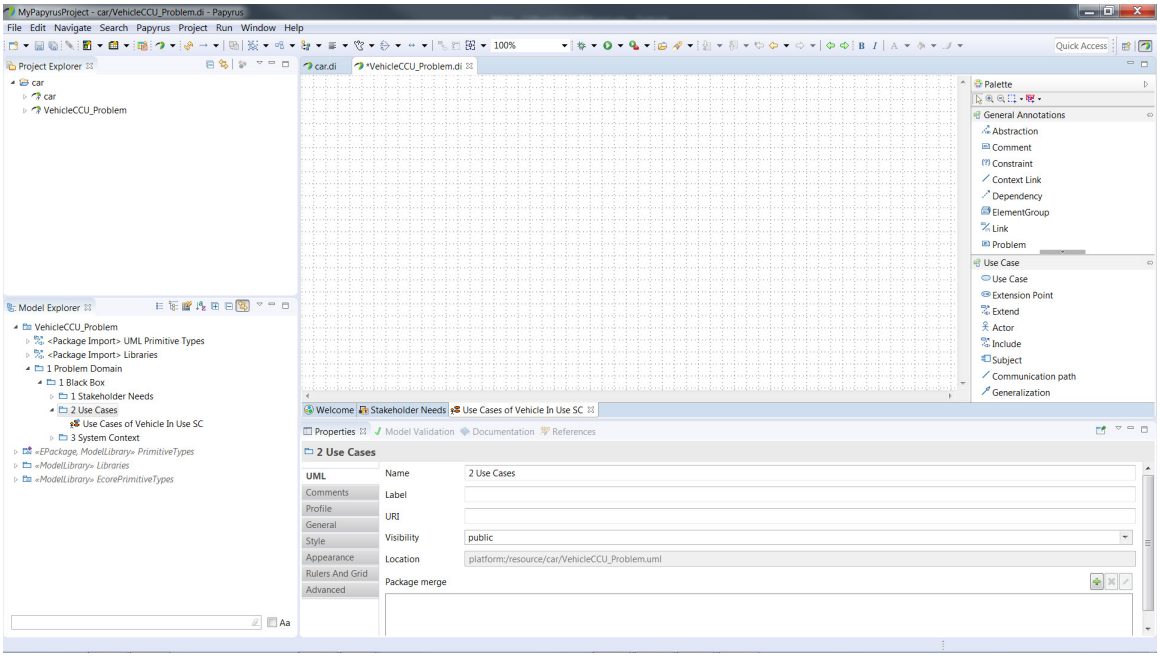

Figure 27: New Use Case Diagram

3. In the *Model Explorer* window, click on the arrow to the left of the *3 System Context* package to expand its contents (figure 28).

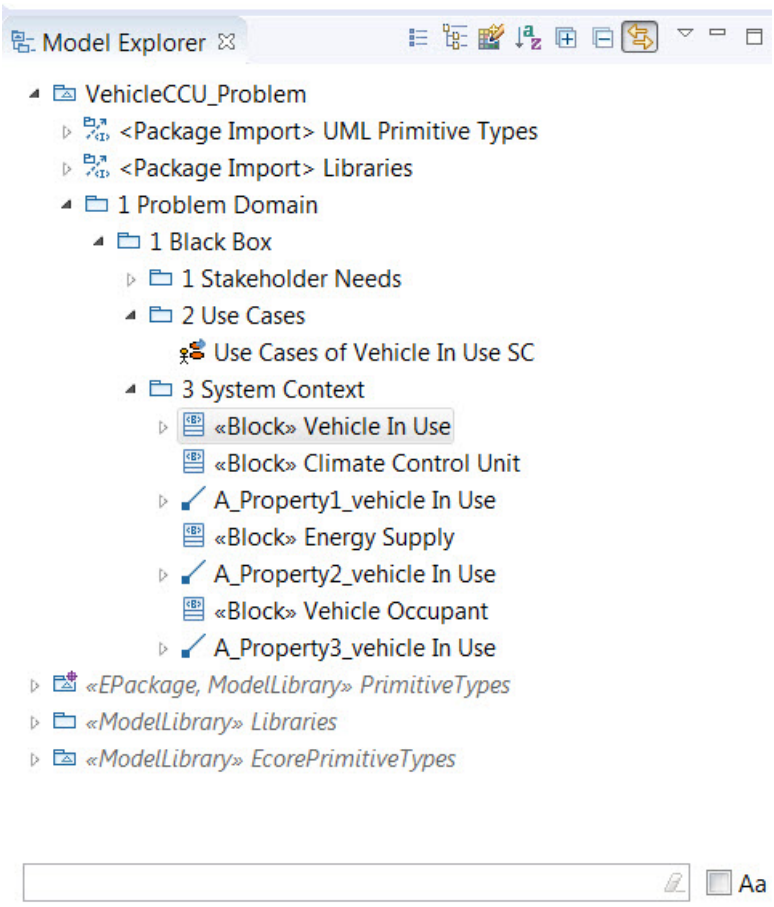

Figure 28: Expand the System Context package

4. Click and Drag the *Vehicle In Use* block onto the *Use Cases of Vehicle In Use* use case diagram in the main window. A copy of the block should appear in the use case diagram (figure 29).

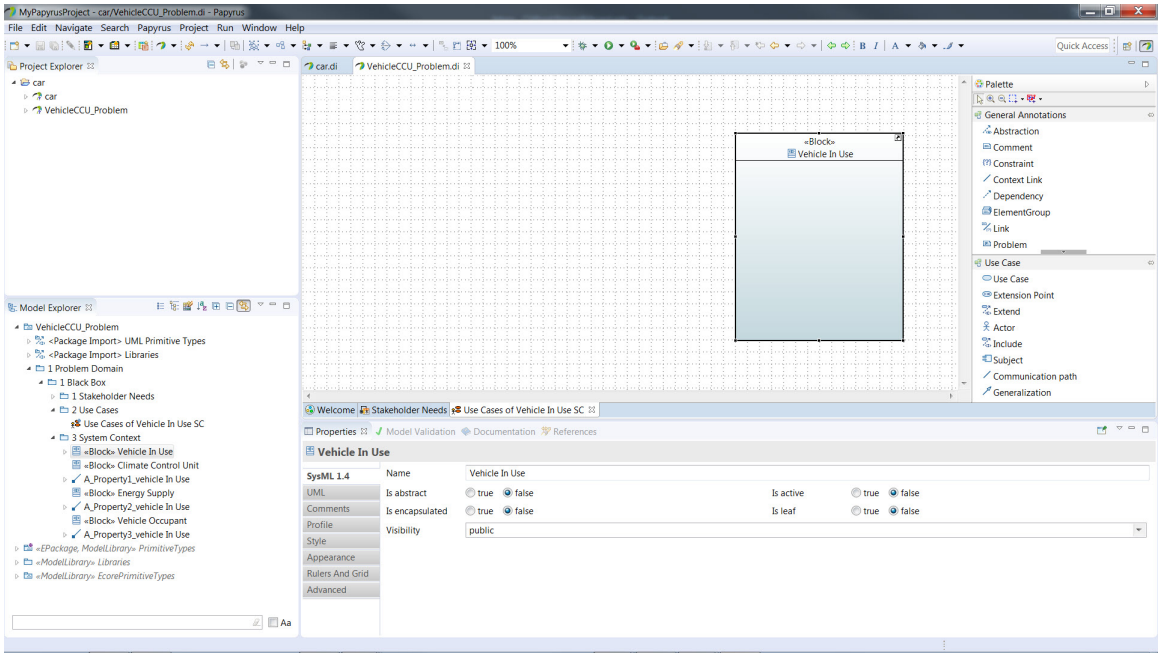

Figure 29: Vehicle In Use block in Use Case Diagram