SysML Hands-On Exercises

Exercise 5.2
SysML Parametrics Diagrams

MagicDraw

August 2018

OBJECTIVES

The objectives of this exercise are to

- Create a flexible Part Property relationship using Multiplicity and Inheritance
- Create a Parametric Diagram using Complex Aggregates
- Create multiple Instances of the Payload assembly
- Solve these Instances for mass roll-up
- Use mass roll-up calculation to verify UAV Payload Mass requirement.

This process is intended to represent building part of a complete mass roll-up calculation for the UAV.

PREPARATION

1. This exercise assumes the student has Cameo System Modeler 19.0 (or MagicDraw with SysML 19.0) and the ParaMagic plug-in for MagicDraw/CSM (ParaMagic 18.0 or later) installed correctly on his or her machine with valid licenses for use.
2. The student should load the Part 5 course materials onto the computer, specifically Exercise 5.2 Starter UAV.mdzip and Exercise 5.2 Final UAV.mdzip.
3. The student should view the video Introduction to SysML Part 5 Exercise 5.2 in its entirety before attempting the exercise.

NOTES AND CAUTIONS

We recommend that the student watch the video demonstration of this exercise in its entirety before beginning their own work. The video includes background and explanatory material that is not repeated in the written instructions.

We also recommend that the student read the material carefully. The most common source of error is confusion between blocks, packages and diagrams, some of which have similar names. When the student is not sure what an element is, either in the browser or in a diagram, select that element and look in the Properties tab for the gray label that identifies the element type. Also, be careful in reading the instructions in realizing when an instruction should be carried out in the browser or in a diagram.

EXERCISE

5.2.1 Start Cameo System Modeler

5.2.2 Open Exercise 5.2 Starter UAV.mdzip

5.2.3 Create a Block Definition Diagram

- Right-click on the **UAV Design** package in the browser.
- Select Create Diagram → SysML Block Definition Diagram.
- Name the diagram **UAV Payload BDD**.
- Drag the following blocks from the **UAV Design** package into the diagram: **Payload**, **Payload Component**, **Radar**, **Thermal_Camera**, **Video_Camera**.
- In the diagram, draw a black diamond arrow from **Payload** to **Payload Component**.
- In the browser, select the new part property under **Payload** and open the Specification window. Change the name to pc and the multiplicity to 0..* as shown in Figure 1.
- In the diagram, using the floating toolbar to add Generalization relationships from **Payload Component to Radar, Thermal _Camera and Video_Camera**.
- The final BDD should appear similar to Figure 2

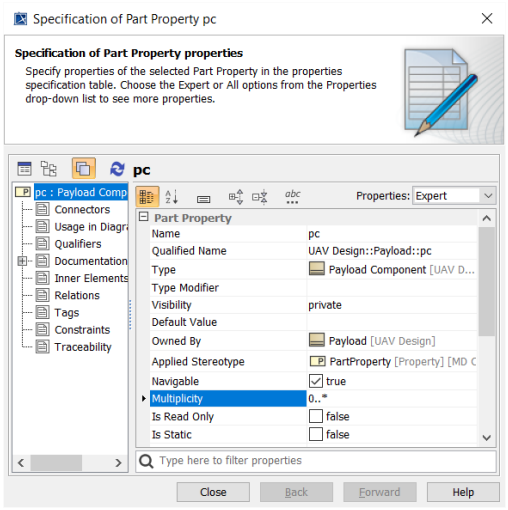

Figure 1 Specification window for part property

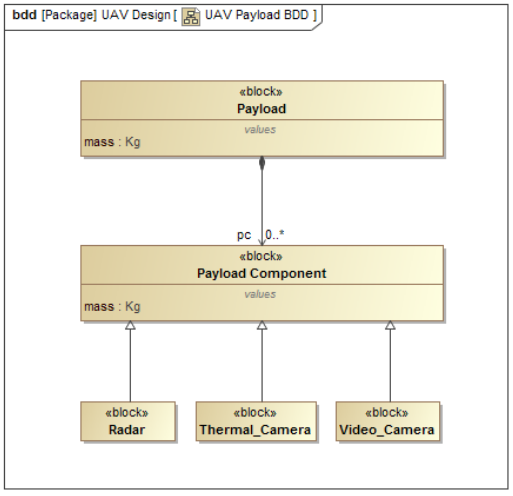

Figure 2 UAV Payload BDD

5.2.4 Create a Parametric Diagram

- Right-click the **Payload** block in the browser. Select Create Diagram → SysML Parametric Diagram.
- Select parts to display as shown in Figure 3.
- Name the diagram **Payload PAR**.
- Drag the constraint block **MassSum** from the **UAV Library::Constraints** package into the diagram. Cancel the Parametric Equation Wizard. We will construct the parametric diagram manually. Name the constraint property created as **ms1** and show the constraint parameters.
- Draw binding connectors as shown in Figure 4.
- Validate the model by selecting the **Payload** block in the browser and clicking the ParaMagic icon at the top menu bar.

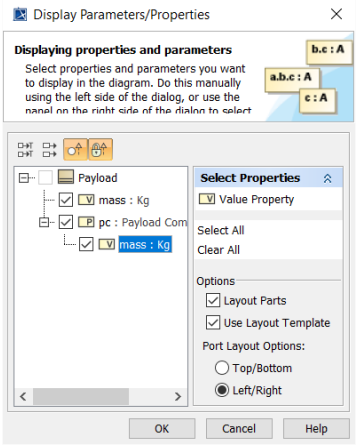

Figure 3 Display parts for parametric diagram

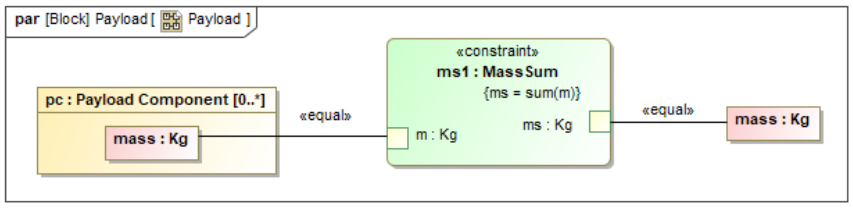

Figure 4 UAV Payload PAR, complete

5.2.5 Create an Instance of the Payload structure

- Right-click the **UAV Analysis** package in the browser.
- Select Create Element → Package. Name the new package **Instance 4.1**.
- Right-click the **Instance 4.1.** package in the browser
- Select Create Element → Instance Specification. Name the new instance **payload 4.1**.
- Select **payload 4.1** and open its Specification window.
- In the first window (Figure 5), click to the right of Classifier in the table, then click the Edit button (highlighted in figure).
- In the second window (Figure 6), start typing in Payload in the search by name field. When the **Payload** block appears below, double-click it to select it and close the window. Close the instance Specification window.
- Repeat this process to create Instances **X512:Radar**, **IR200:Thermal Camera**, and **Minicam:Video Camera**.
- Right-click on the **UAV Analysis::Instance 4.1** package in the browser.
- Select Create Diagram → SysML Block Definition Diagram.

- Name the diagram **Instance 4.1 BDD**. Drag all the instances into the diagram**.** The final diagram should appear similar to Figure 7.

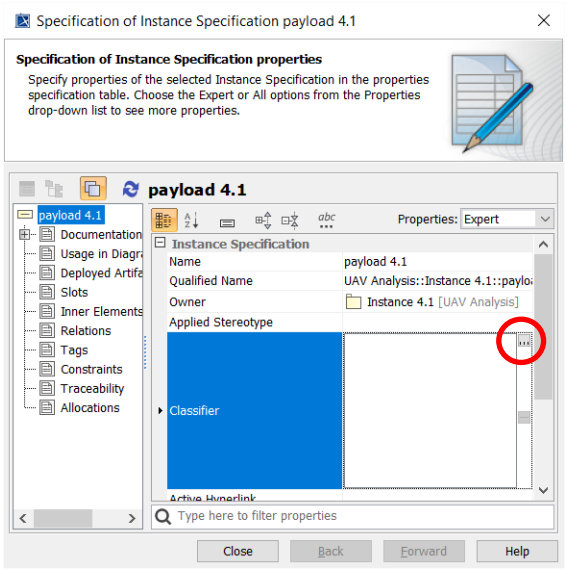

Figure 5 Specifying Instance, stage 1

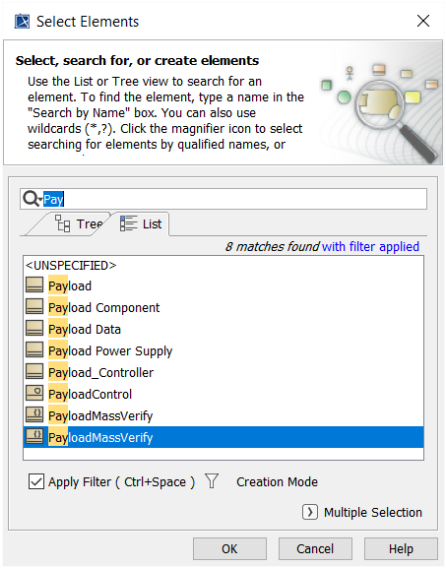

Figure 6 Specifying Instance, stage 2

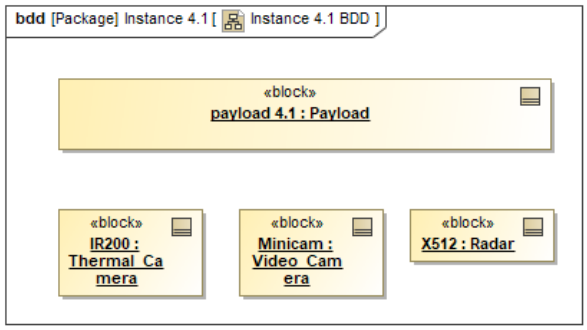

Figure 7 Instance 4.1 BDD, stage 1

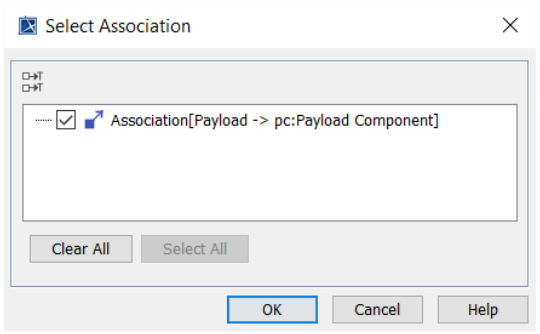

Figure 8 Select Association, first screen

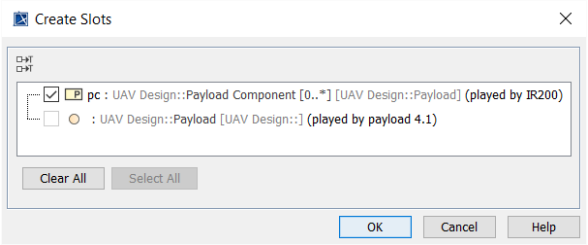

Figure 9 Select Association, second screen

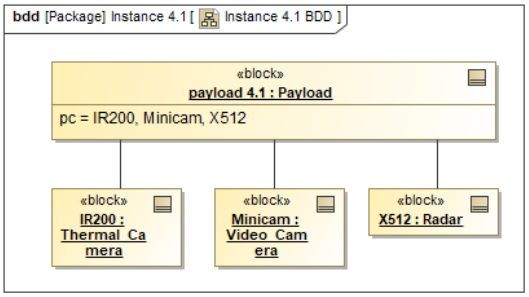

Figure 10 Instance 4.1 BDD, final

- Create link from the payload instance to each of the payload component instances.
  - Select **payload 4.1**. On the floating toolbar, click the link icon (line with a small L)
  - Drag the end of the link to **X512:Radar** and click to set.
  - In the first window (Figure 8), accept the default and click OK.
  - On the second window (Figure 9), accept the default and click OK.

- Repeat this process for links to the thermal camera and video camera instances.
- In the browser, select **payload 4.1:Payload** and click the ParaMagic icon in the menu bar.
- The first error message says that one or more slots in the instance model are not initialized. Click Initialize.

5.2.6 Assign Initial Values and Causalities in ParaMagic

- Click the Expand button to show all parameters. In the browser, set the initial values and causalities as follows
  - Payload 4.1:Payload causality = target
  - X512:Radar Causality = given Value = 20
  - IR200:Thermal Camera Causality = given Value = 20
  - Minicam:Video Camera Causality = given Value = 10
- The result (Figure 11) can be used to update the SysML instance, if desired.
- Click Solve. The final result is shown in Figure 12.

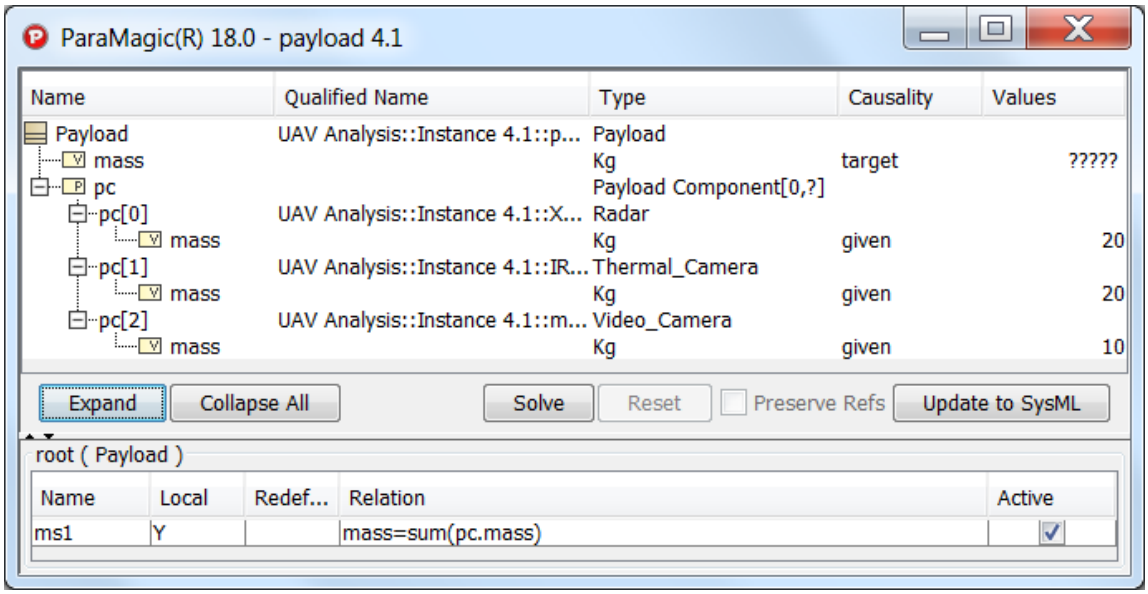

Figure 11 ParaMagic browser, stage 1

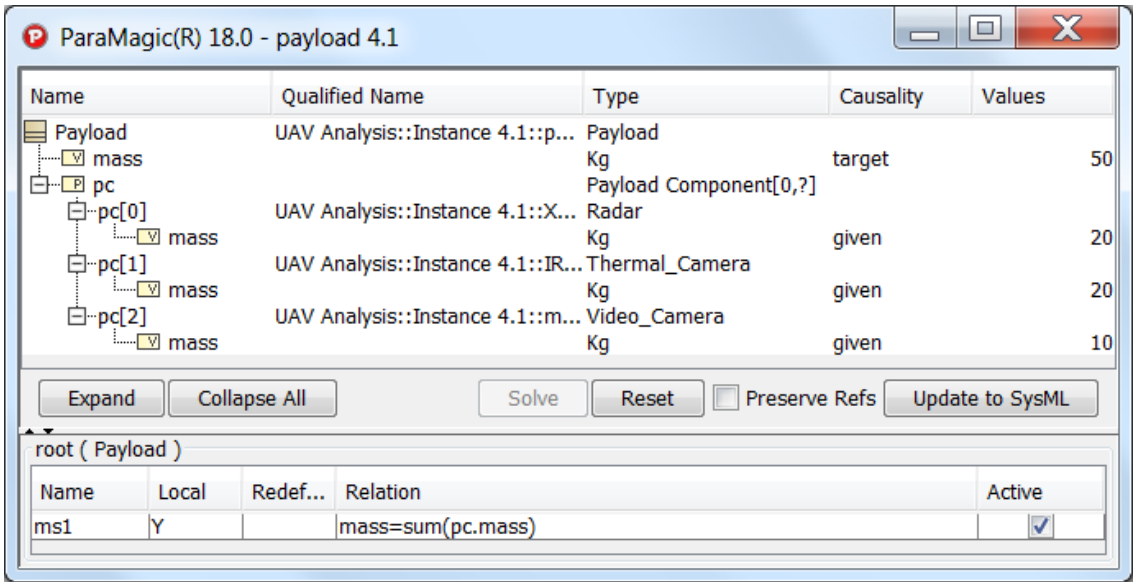

Figure 12 ParaMagic browser, stage 2

5.2.7 Create a Second Instance of the UAV structure

- Right-click the **Instance 4.1** package in the browser and select Copy.
- Right-click the **UAV Analysis** package in the browser and select Paste.
- Edit the name of the payload instance to **payload 4.2:Payload** and the name of the diagram to **Instance 4.2 BDD**.
- In the browser, delete the instances **X512:Radar** and **IR200:Thermal Camera**.
- Right-click **minicam:Video Camera** in the **Instance 4.2** package and select Copy.
- Right-click the **Instance 4.2** package in the browser and select Paste.
- Drag the new instance, **minicam1:Video Camera**, into the diagram **Instance 4.2 BDD.**
- Create a link from **payload 4.2:Payload** to **minicam1:Video Camera**.
- The final instance diagram should appear similar to Figure 13.
- In the browser, select **payload 4.2:Payload** and click the ParaMagic icon in the menu bar. The browser should look similar to Figure 14.

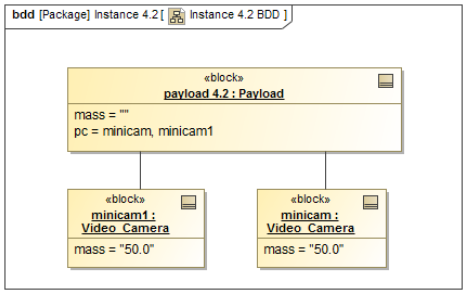

Figure 13 Instance 4.2 BDD

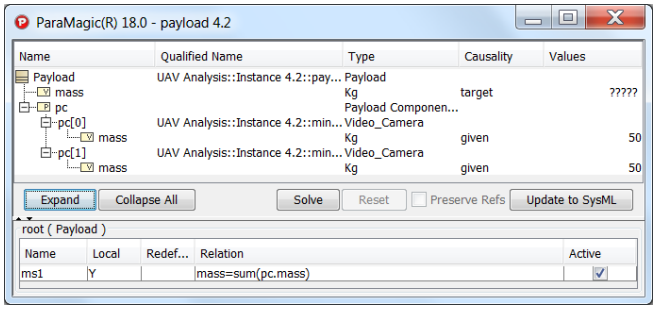

Figure 14 ParaMagic browser, Instance 4.2

5.2.8 Add Requirement Verification to the Mass Roll-up model

- Right-click the **Payload** block in the browser and select Create Element → Value Property.
- Name the new value property **massVerdict** and set the valuetype to **Real**.

- Drag **massVerdict** into the parametric diagram created earlier, **Payload PAR**.

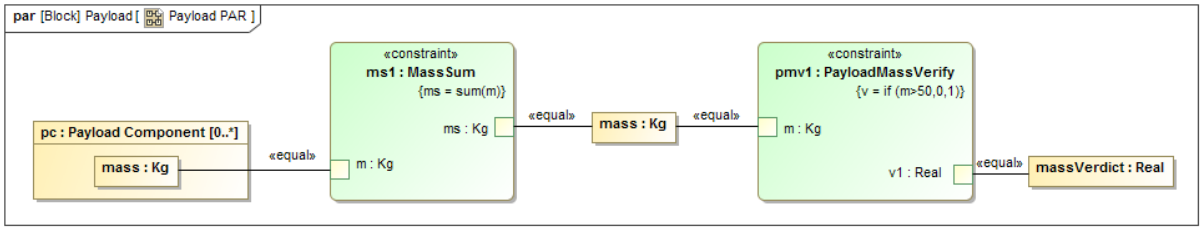

Figure 15 Payload PAR after requirements verification added.

- Locate the constraint block **PayloadMassVerify** inside the **UAV Requirements::Related Elements::Constraints** package and drag it into the parametric diagram. Name the constraint property created **pmv1**.
- Add binding connectors. The final diagram should appear similar to Figure 15.
- Use ParaMagic to validate **Payload** block.

5.2.9 Modify Instances and Solve

- In the browser, select the **payload 4.1:Payload** instance and launch the ParaMagic browser.
- Initialize and/or reassign causality when requested.
- The ParaMagic browser before solving should look like Figure 16.
- When finished, save and close the project.

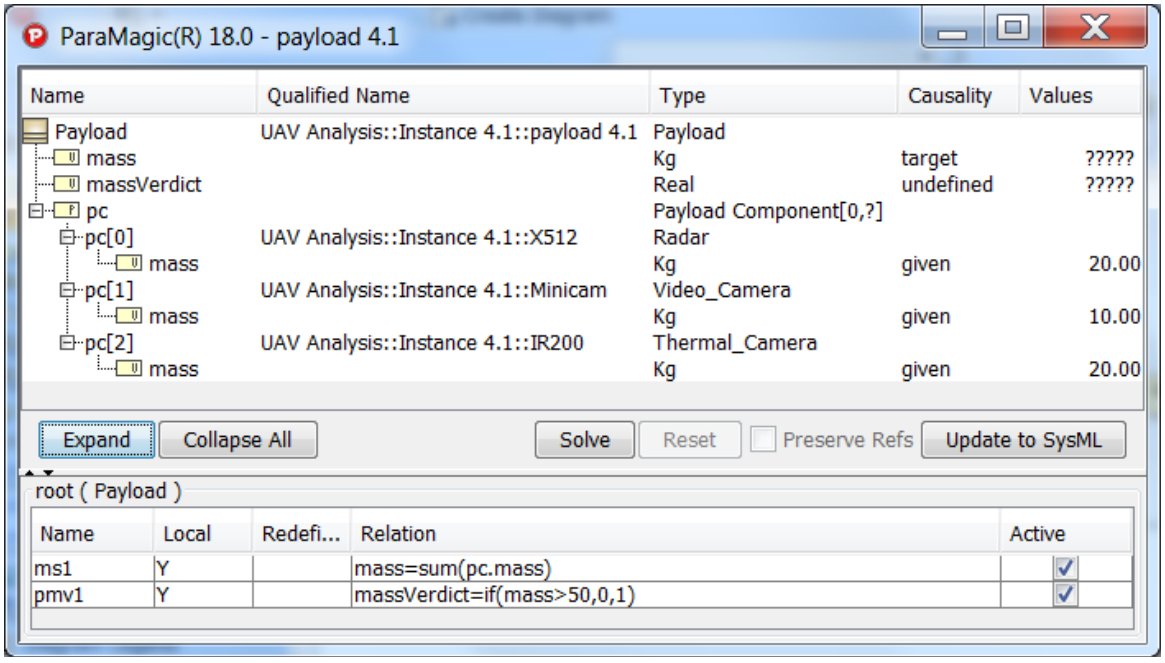

Figure 16 ParaMagic browser, Instance 4.1, after modification