

# Rollup Pattern simulation

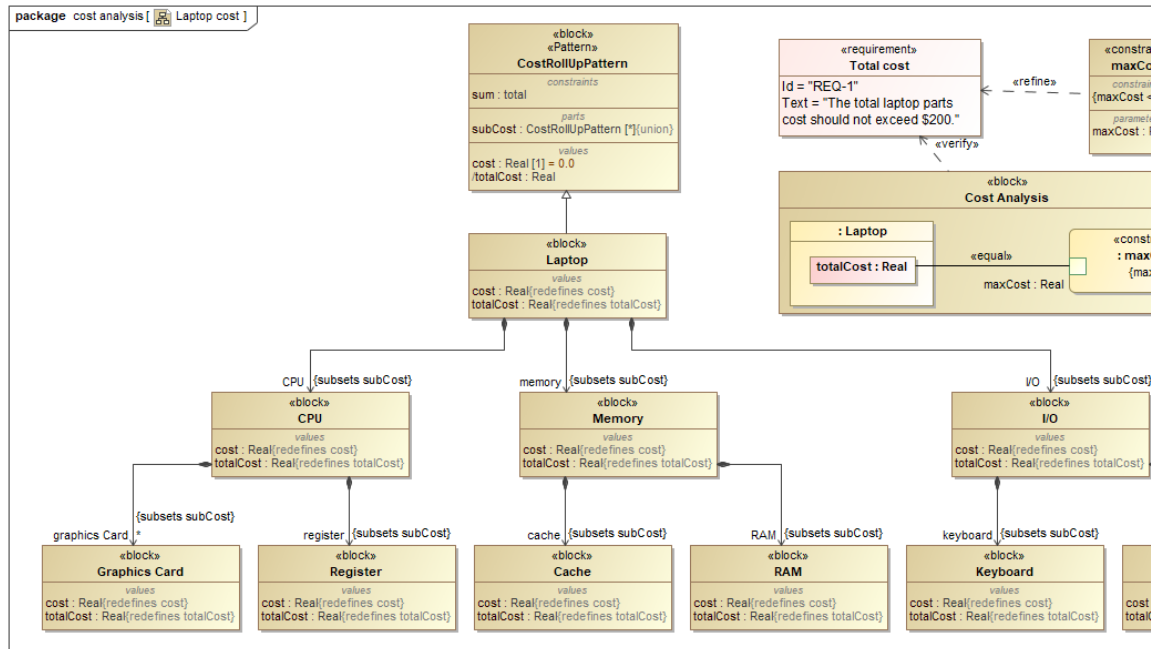
On this page

- [Simulation with a built-in rollup pattern](#)
- [Simulation with a custom rollup pattern](#)

Cameo Simulation Toolkit supports rollup calculations of total mass, cost, power, and another system dimension, based on individual values of all the parts in the model. Please refer to [Rollup Pattern Wizard](#), [applying Rollup Pattern Blocks](#), and **LaptopCostAnalysis** and **SpacecraftMassRollup** built-in samples for more details.

## Simulation with a built-in rollup pattern

The **LaptopCostAnalysis** sample applies built-in **CostRollUpPattern**. The sample uses the **Cost Analysis** Block with the **Laptop** context Block as the Part. Optionally, you can add any constraint (*maxCost*) and Requirement (*Total cost*) and use a Parametric diagram to connect them as shown in the figure below.



The Cost Analysis Block containing the Laptop context Block applied with built-in CostRollUpPattern as the Part.

You can also either run the simulation through the **Cost Analysis** Block directly or use a **SimulationConfig** with `executionTarget = the Cost Analysis Block`. The result will be shown in the **totalCost** context Block, e.g., 220 (80+50+90) as shown in the figure below.

Simulation

Simulation

Variables

Sessions

Console

Breakpoints

Name

Value

Cost Analysis

analysis : Cost Analysis@5e19af...

Laptop

laptop : Laptop@6b314543

cost : Real

0.0000

totalCost : Real

220.0000

CPU : CPU {subsets subCost}

cpu : CPU@21080ae8

cost : Real

80.0000

totalCost : Real

80.0000

gcard : Graphics Card [\*] {subsets subC...

register : Register@4e3100db

I/O : I/O {subsets subCost}

io : I/O@7e981cf6

cost : Real

0.0000

totalCost : Real

50.0000

keyboard : Keyboard {subsets subCost}

keyboard : Keyboard@281fc64f

touchpad : Touchpad {subsets subCost}

touchpad : Touchpad@a9b895

cost : Real

50.0000

totalCost : Real

50.0000

memory : Memory {subsets subCost}

memory : Memory@4b9179d9

cost : Real

50.0000

totalCost : Real

90.0000

cache : Cache {subsets subCost}

cache : Cache@5b4f2e96

RAM : RAM {subsets subCost}

ram : RAM@213e1de

cost : Real

40.0000

totalCost : Real

40.0000

Results

Criteria

Classifier: CostRollUpPattern

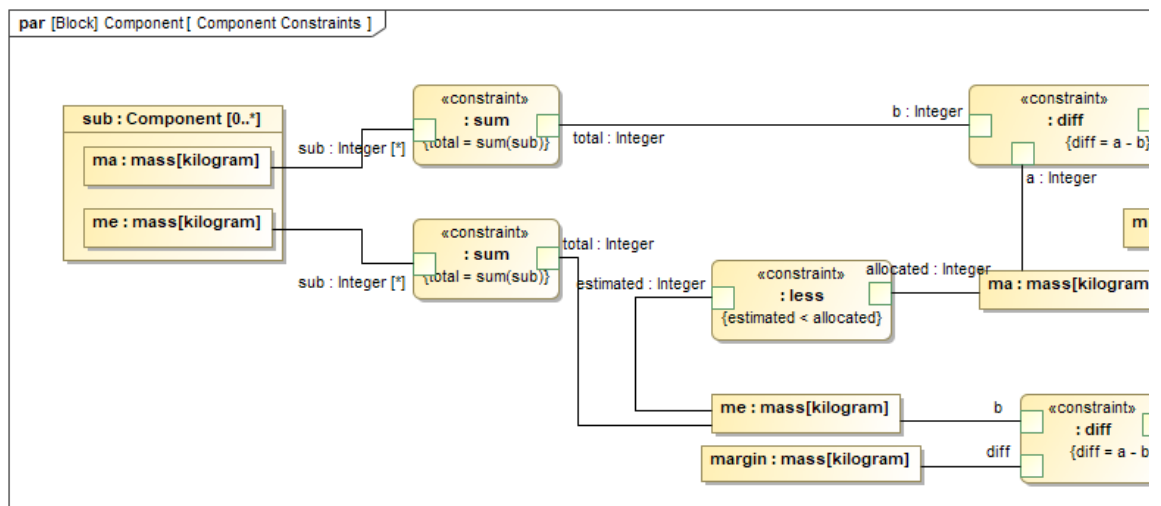
Scope (optional)

| #  | Name     | Classifier    |
|----|----------|---------------|
| 1  | analysis | Cost Analysis |
| 2  | laptop   | Laptop        |
| 3  | io       | I/O           |
| 4  | keyboard | Keyboard      |
| 5  | touchpad | Touchpad      |
| 6  | memory   | Memory        |
| 7  | ram      | RAM           |
| 8  | cache    | Cache         |
| 9  | cpu      | CPU           |
| 10 | register | Register      |
| 11 | gcard    | Graphics Card |

The totalCost context Block as the result of Laptop cost analysis applied with CostRollUpPattern.

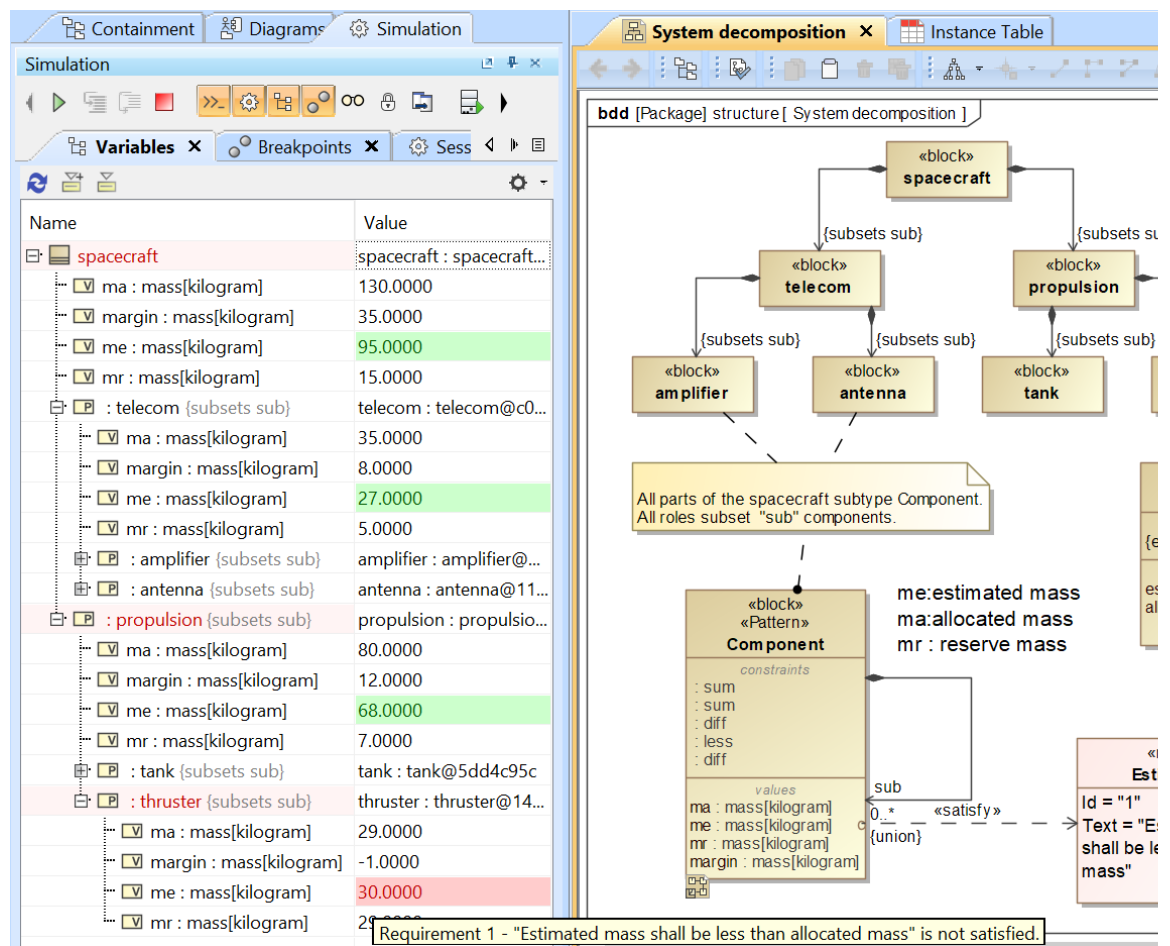
## Simulation with a custom rollup pattern

The **SpacecraftMassRollup** sample applies a custom rollup pattern through the same [applying Rollup Pattern Block](#) like the built-in one but requires selecting a Pattern Block for the custom rollup pattern, e.g., **Component**. The **Component** Pattern Block uses value type properties binding with 5 constraint properties in the Parametric diagram as shown below.



A Parametric diagram of the Component custom rollup pattern as a Pattern Block.

Optionally, you can add any constraint (*less*) and Requirement (*Estimated mass*), and then run the *spacecraft mass analysis* [SimulationConfig](#) with the *executionTarget* as an instance of the spacecraft to run. The result of evaluation, according to the *me < ma* rule, will be highlighted at *me : mass[kilogram]* for each component shown on the right side in the Block Definition diagram as shown below.



The evaluation result (me : mass[kilogram]) of the SpacecraftMassRollup sample applied with the Component custom rollup pattern.