SysML Miscellaneous Grab Bag

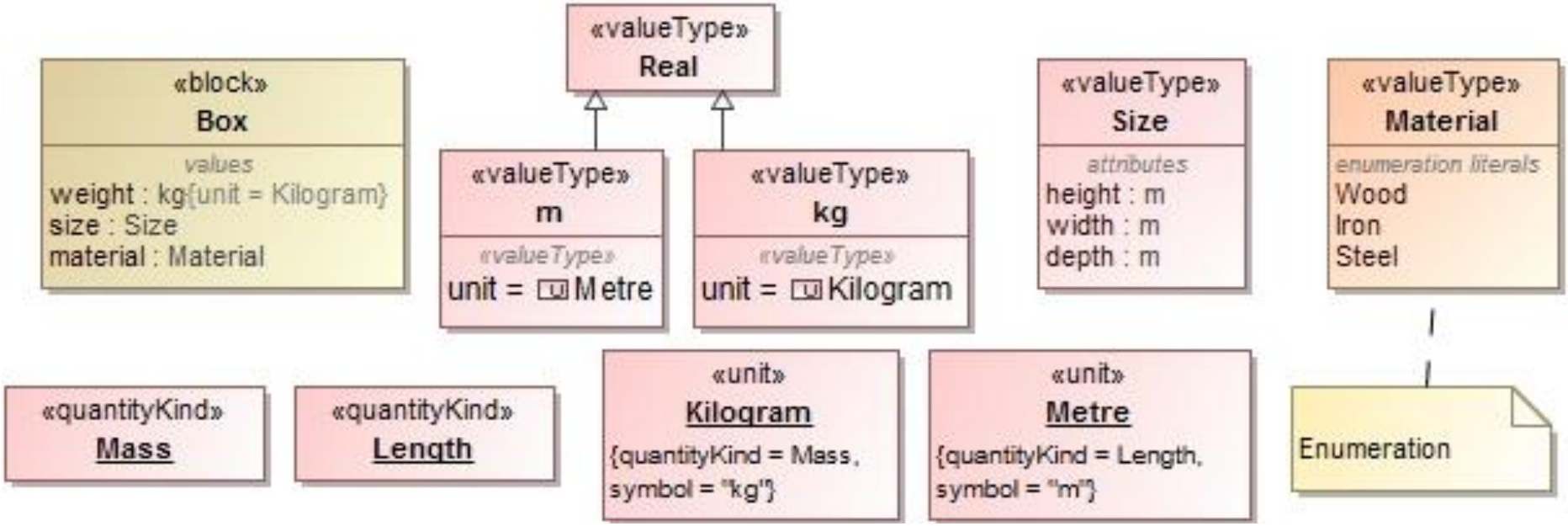
Systems Engineering BSc Course

Modeling of logical and physical data

Using block definition diagrams

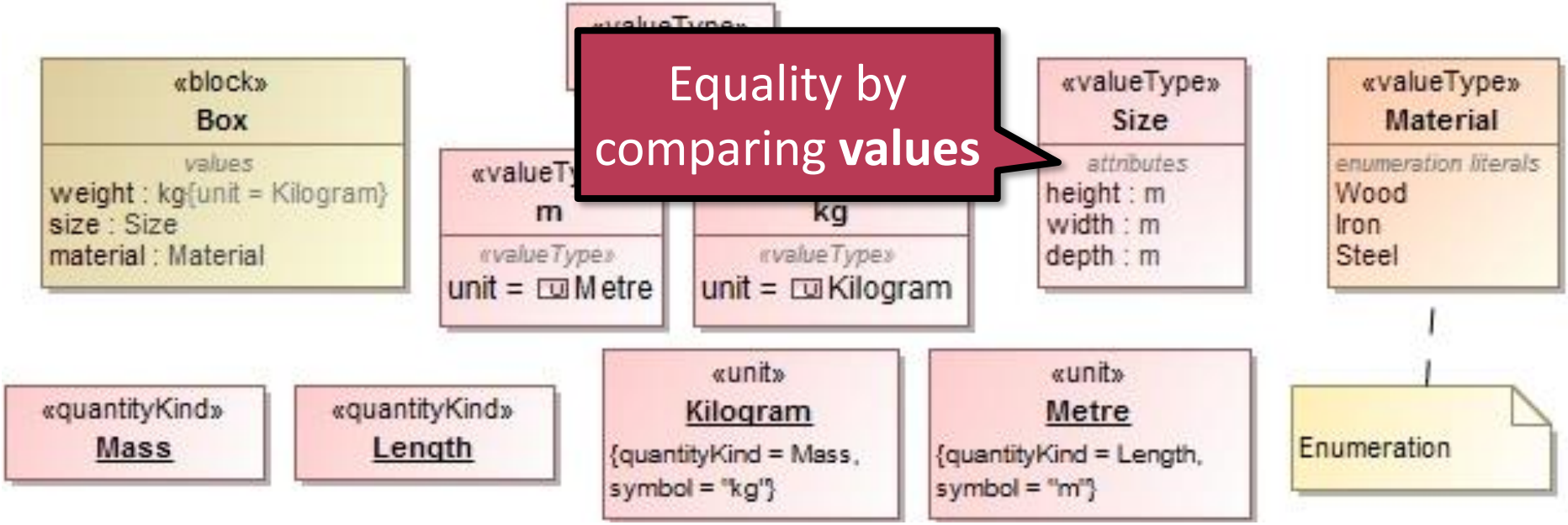
Value type (Data type)

- Primitives: Boolean, String, Complex, etc.
- Can have Unit and/or QuantityKind (formerly dimension)
  - QuantityKind: Length, Energy, Time, etc.
  - Unit: meter, inch, Watt, secundum, etc.
    - Has a QuantityKind



Value type (Data type)

- Primitives: Boolean, String, Complex, etc.
- Can have Unit and/or QuantityKind (formerly dimension)
  - QuantityKind: Length, Energy, Time, etc.
  - Unit: meter, inch, Watt, secundum, etc.
    - Has a QuantityKind

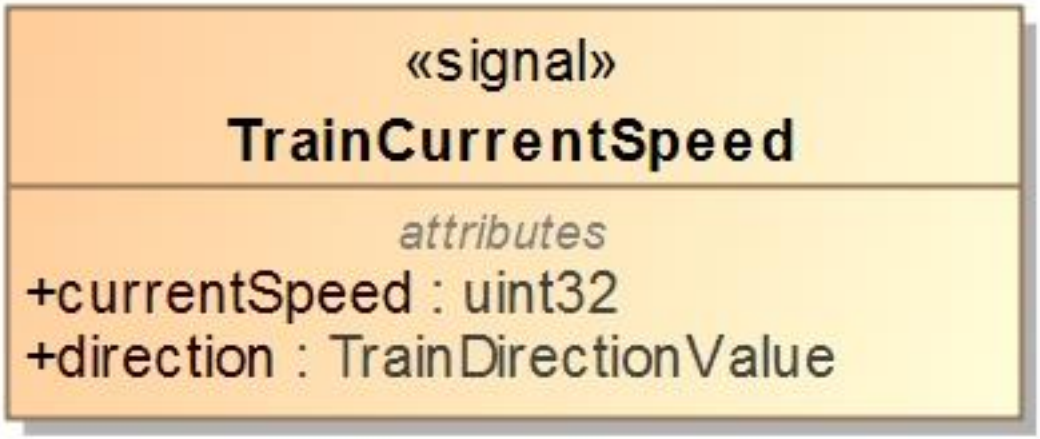


Data of a block

- Blocks can have attributes and/or values
  - Part property (composition)
  - Reference property
  - **Value property (value type)**
  - Flow property
- Value given by / restricted by
  - Definition (bdd)
    - e.g. in a specialized block (motorized =„true”)
  - Use (ibd)
  - Runtime
    - The value may change over time

Signal, Block

- A **signal** defines a message that can be sent and received by a block.
  - Has a set of attributes
  - Used by interfaces in ***Signal Receptions***



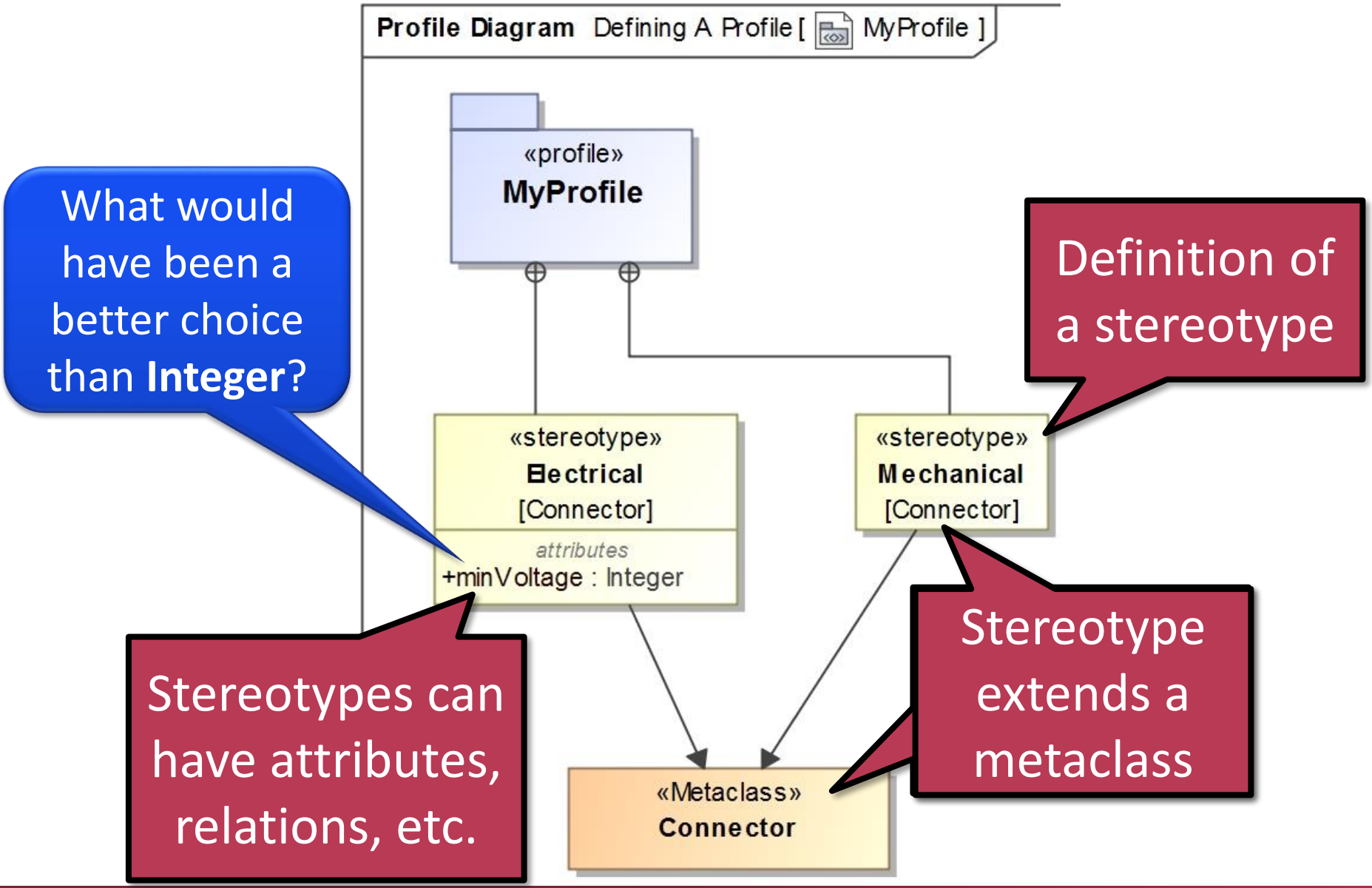
Profiles

for extending UML/SysML

UML Profiles

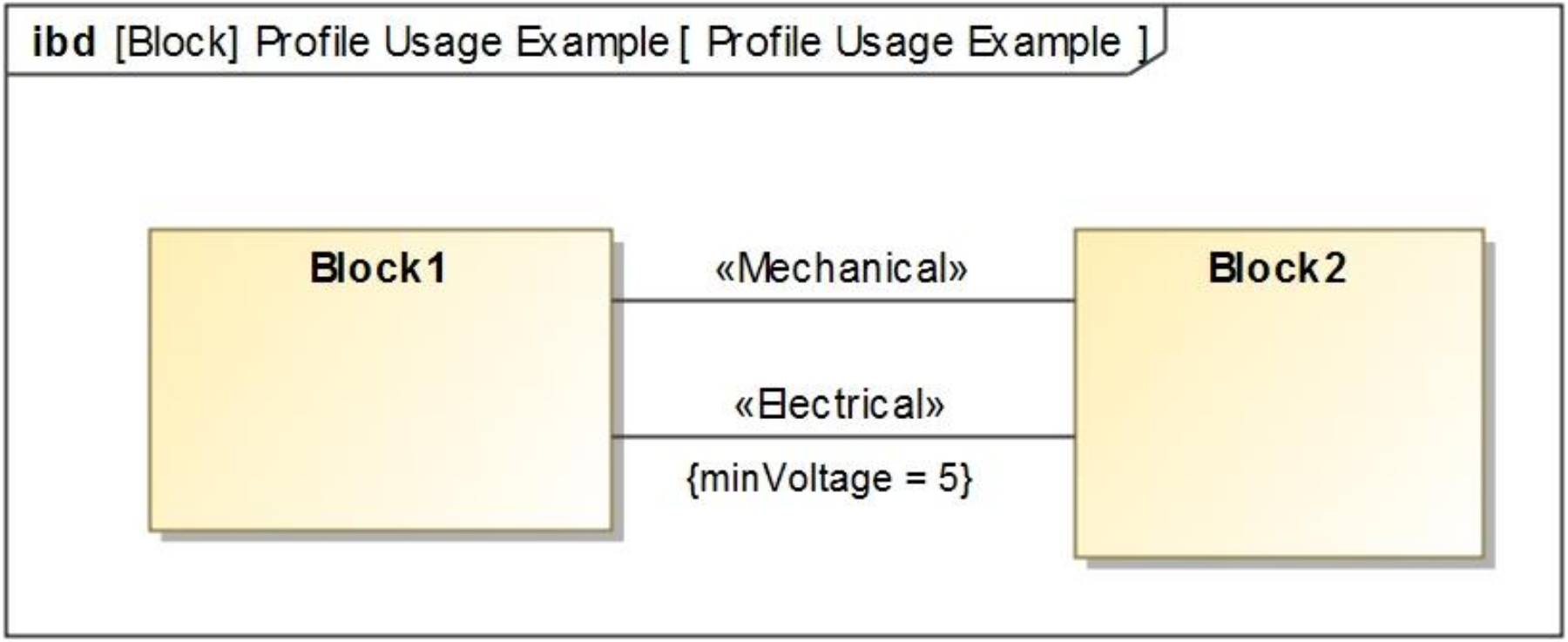
- Profiles can be used to extend the UML/SysML language.
- Examples
  - SysML is defined as a profile on a subset of UML.
  - SYSMOD (a methodology for SysML) also defines a profile for SysML
  - MARTE (which is an OMG standard) profile is used for modeling real-time and embedded applications.
  - SysPhS (also OMG) is an add-on to SysML for modeling physical interaction and signal flow
  - Tools usually support the creation of custom profiles.

Defining a Profile



Using a Profile

- A profile should be applied to the project to use

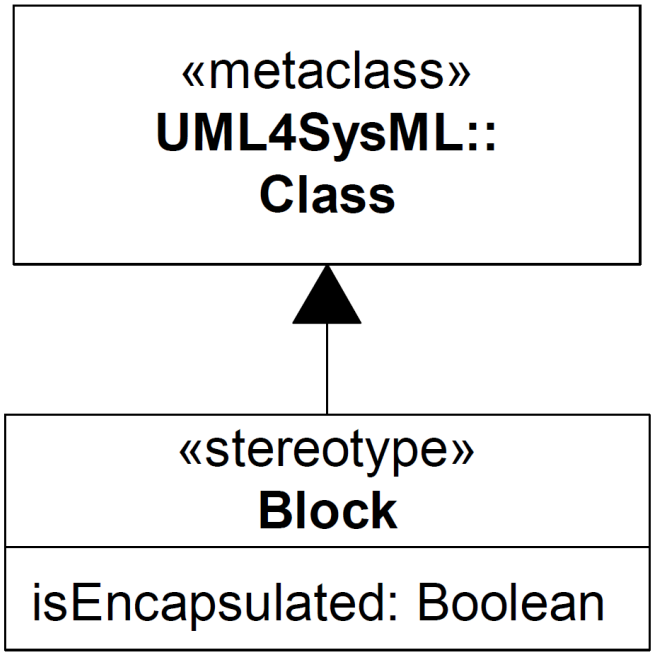


What else may a Profile provide?

- Presentation elements: icons, diagrams, …
- Documented meaning
  - “A «Mechanical» Connector it is expected to exert significant forces in resistance to displacement of…”
- Design rules that can be enforced by the editor
  - E.g. “if a «Device» Block has an «Electrical» Connector, it must also own a «Ground» Port”
  - We will return to “well-formedness constraints” soon
- External tools that understand the Profile
  - Simulators, code generators, analysis tools etc.

SysML as a Profile

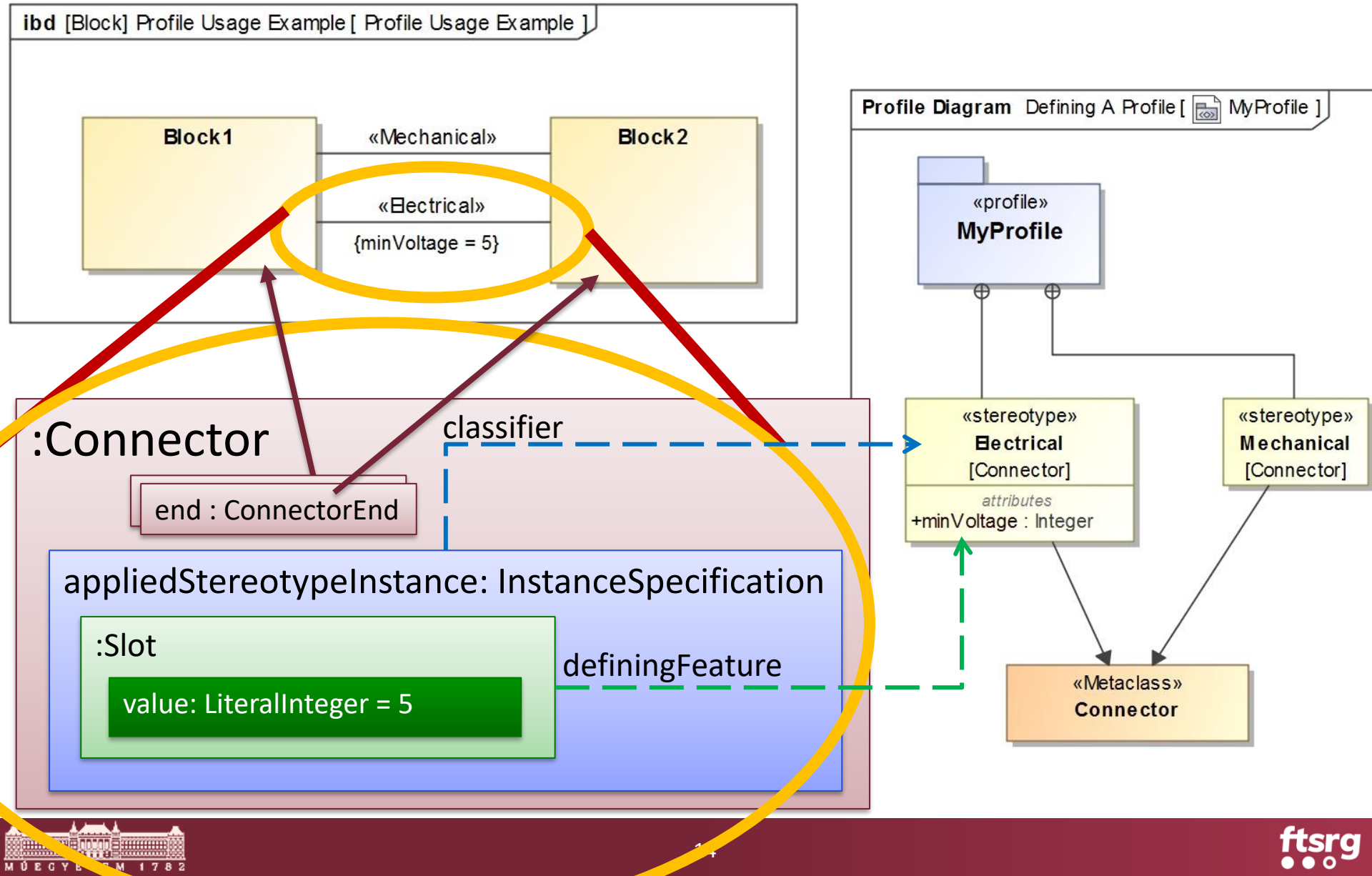
- SysML itself is a Profile on UML



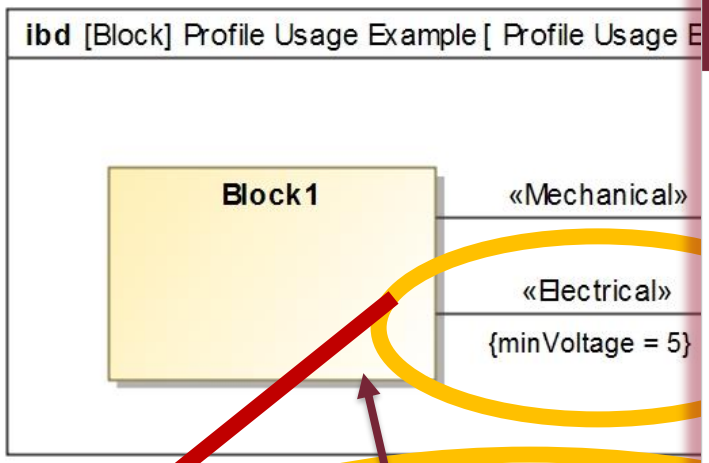
UML in Abstract Syntax

The underlying infrastructure of UML and SysML

Under the hood, pt. 1



Under the hood, pt. 2



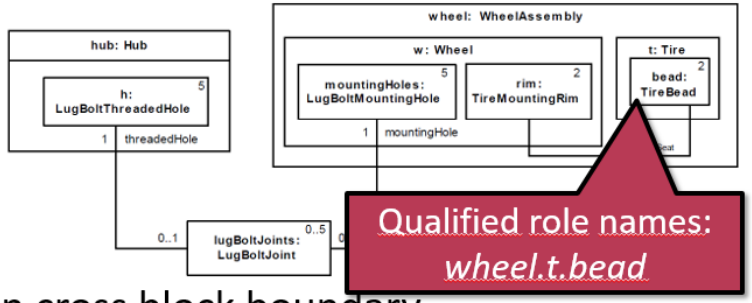
:Connector

end : ConnectorEnd

NOTE: the "ConnectorEnd" objects are actually oversimplified here… let's recap why!

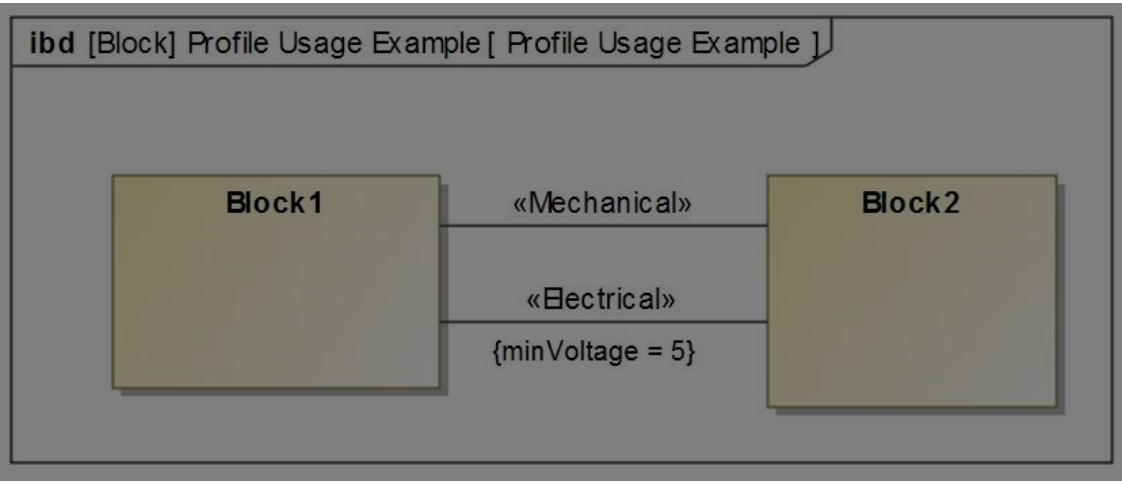
Nested blocks

- Nested blocks
  - Block structure is expanded in an embedded **ibd**
  - Commonly used on **ibd**s
    - (Sometimes on **bdd**, in the *structure* compartment)



- Encapsulation
  - Connectors can cross block boundary
  - Mark the block *encapsulated* to forbid this

Abstract and Concrete Syntax

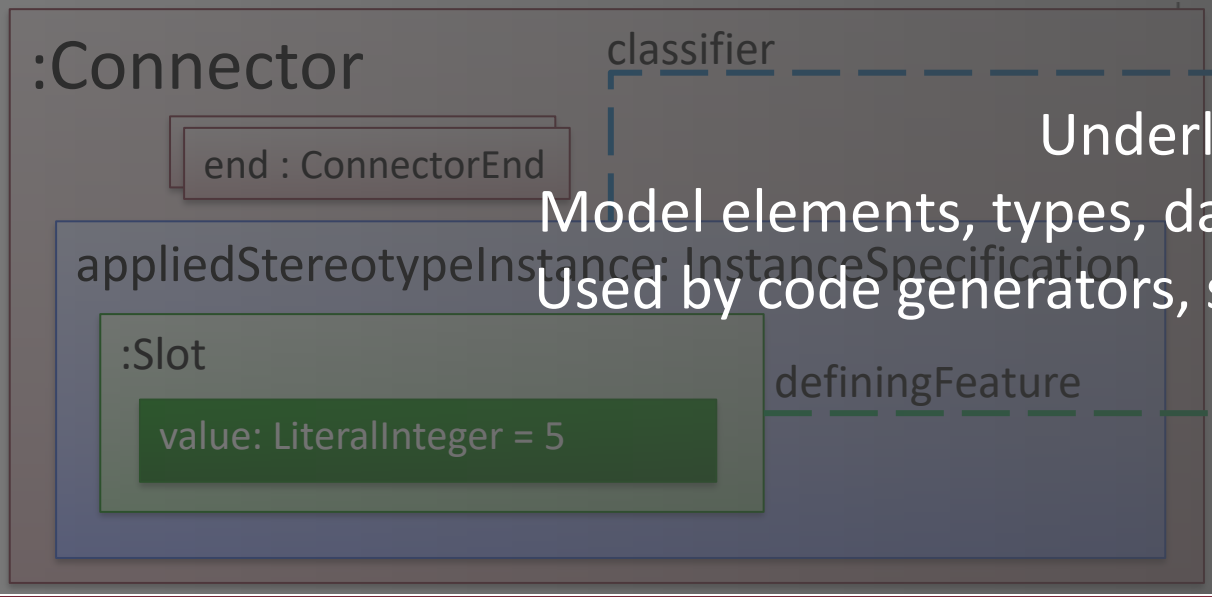


Concrete Syntax

User-friendly presentation

Diagrams, tables, icons

Displayed in editors, views



Abstract Syntax

Underlying **graph** representation

Model elements, types, data fields, cross-references

Used by code generators, simulators, model analysis

Well-formedness constraints

Types of constraints

- A property of the **system**
- Must hold at any point in time
- "If you realize this system, we assume …
  (you must guarantee it, or must be a law of nature)"

- System Constraints
  - CPU should receive 12V +- 1V electricity
  - Dissipated heat equals current times voltage

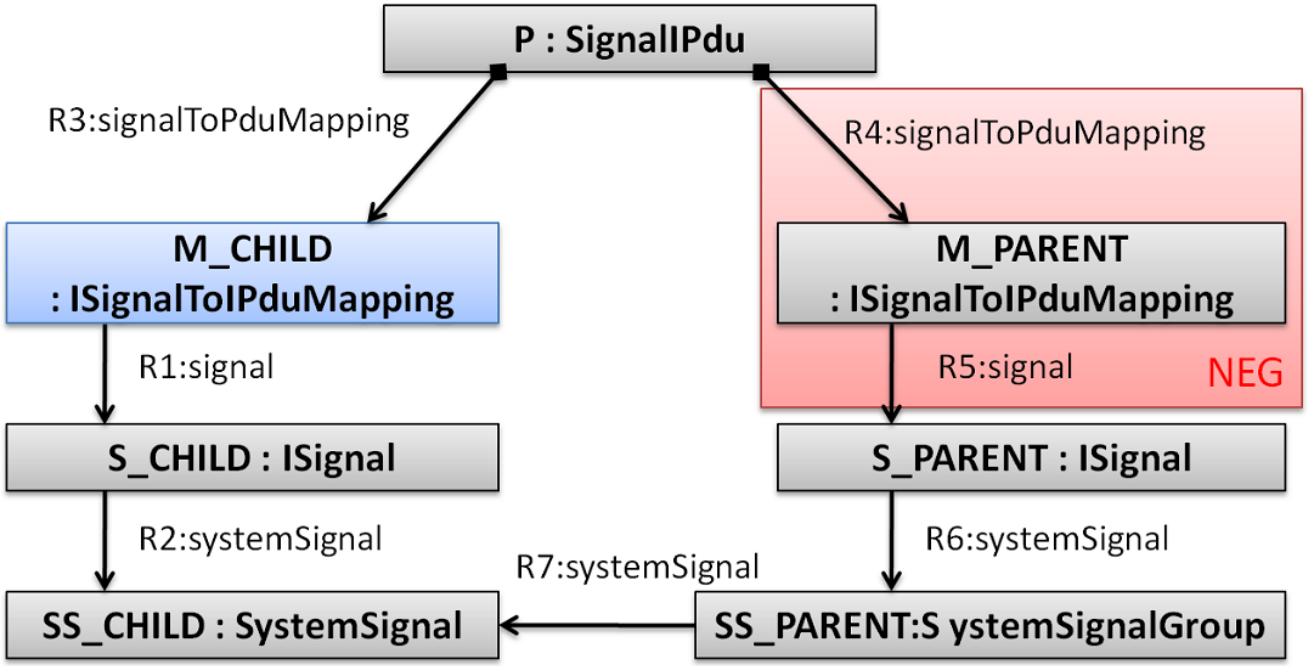
- A property of the **model**
- Can be **enforced** by the editor
- "If a model uses this element, it must …"

- Well-formedness Constraints / Design Rules
  - "if a «Device» Block has an «Electrical» Connector, it must also own a «Ground» Port"

Motivation: Early validation of design rules

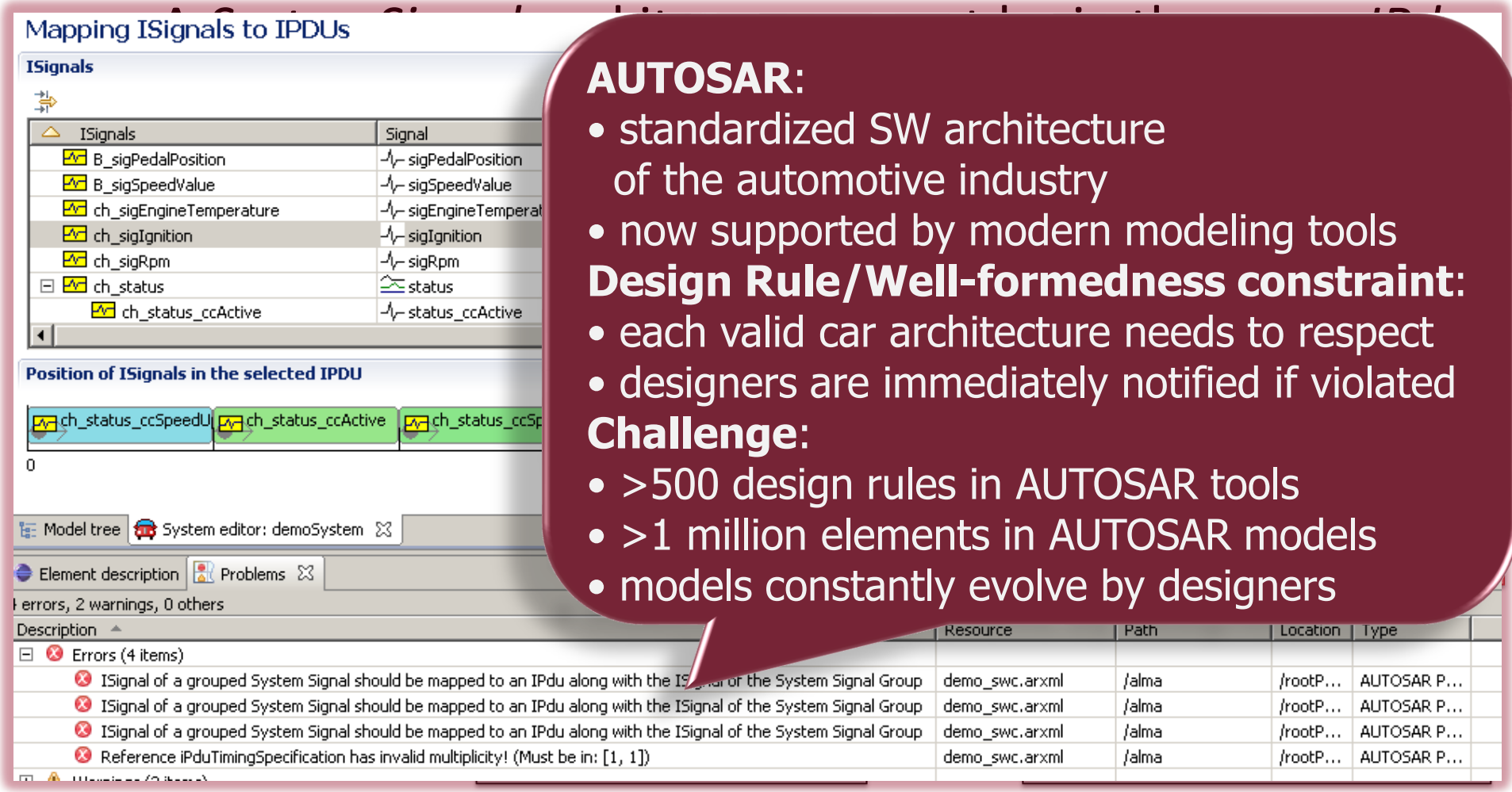
SystemSignalGroup design rule (from AUTOSAR)

- A *SystemSignal* and its group must be in the same *IPdu*
- Challenge: find **violations** quickly in large models
- New difficulties
  - reverse navigation
  - complex manual solution



Motivation: Early validation of design rules

SystemSignalGroup design rule (from AUTOSAR)



AUTOSAR:

- standardized SW architecture of the automotive industry
- now supported by modern modeling tools

Design Rule/Well-formedness constraint:

- each valid car architecture needs to respect
- designers are immediately notified if violated

Challenge:

- >500 design rules in AUTOSAR tools
- >1 million elements in AUTOSAR models
- models constantly evolve by designers

OCL: an OMG Standard

- Object Constraint Language
  - OMG standard
  - Declarative language for defining constraints
  - ~ functional programming

- Unique name constraint defined by OCL:
  - ```
    context Component inv:
      Component.allInstances()->
      forAll(c1, c2 |
            c1 <> c2 implies c1.name <> c2.name)
    ```
    ```

VIATRA

- VIATRA is an open source Eclipse project
  - Affiliated with the research group
- VIATRA Query Language
  - Graph pattern matching
  - Can evaluate queries incrementally upon changes
- Unique name constraint defined by VQL

```
pattern nameCollision(c1, c2){
  Component.name(c1,name);
  Component.name(c2,name);
  c1 != c2;
}
```

System constraints and physical parameters in SysML

Constraint blocks

Types of constraints

- A property of the **system**
- Must hold at any point in time
- "If you realize this system, we assume …
  (you must guarantee it, or must be a law of nature)"

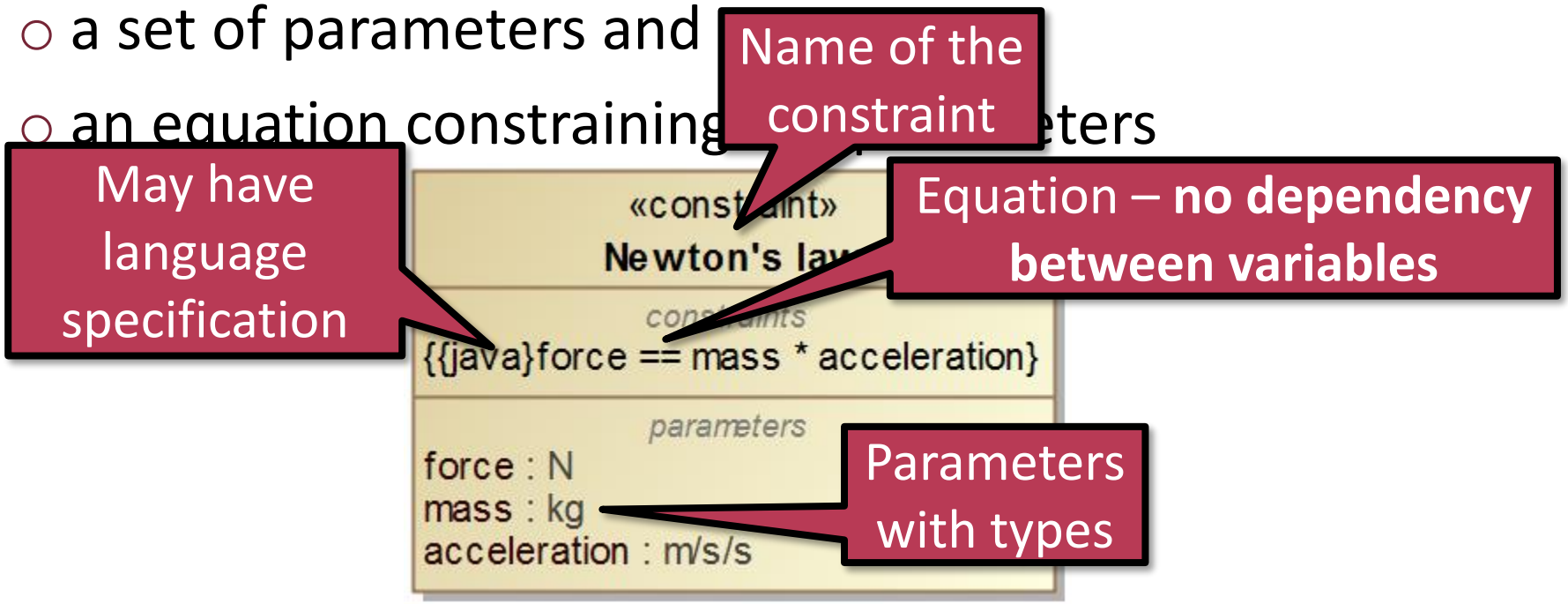
- System Constraints
  - CPU should receive 12V +- 1V electricity
  - Dissipated heat equals current times voltage

- A property of the **model**
- Can be **enforced** by the editor
- "If a model uses this element, it must …"

- Well-formedness Constraints / Design Rules
  - "if a «Device» Block has an «Electrical» Connector, it must also own a «Ground» Port"

Constraint blocks

- **Constraint**: equations with parameters bound to the properties of the system
- **Constraint block**: supports the definition and the reuse of constraints. It holds
  - a set of parameters and
  - an equation constraining [illegible]eters



Assignments and equations

- **Causal** connection ≈ assignment in programming language

$$y := x + 3$$

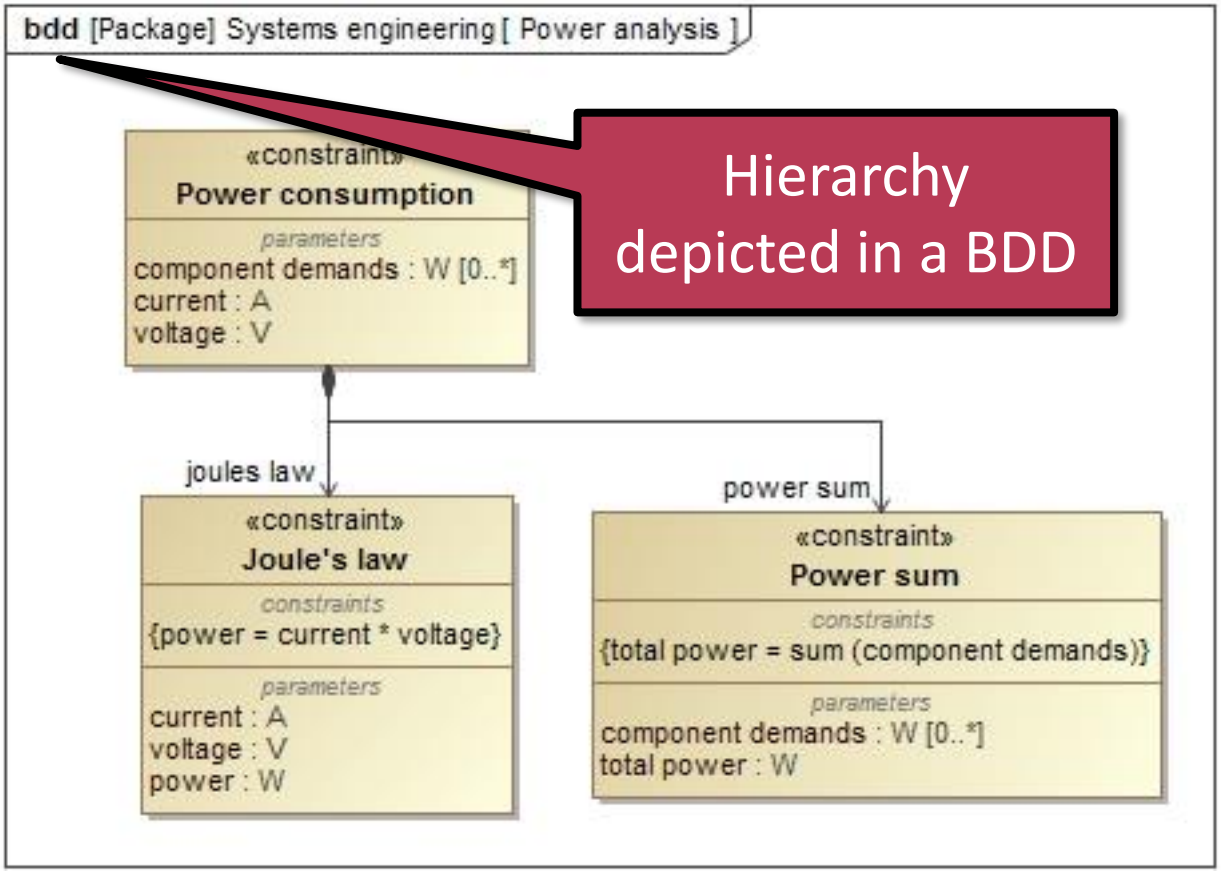
  - Right-hand-side value determines left-hands-side variable
  - Typical use: to implement controller
- **Acausal** connection ≈ mathematical equation

$$y = x + 3 \Leftrightarrow y - 3 - x = 0$$

  - Always holds; if any of the variables has a new value, it enforces that the other variables change accordingly
  - Typical use: to model behaviour of plant / environment

Constraint definition

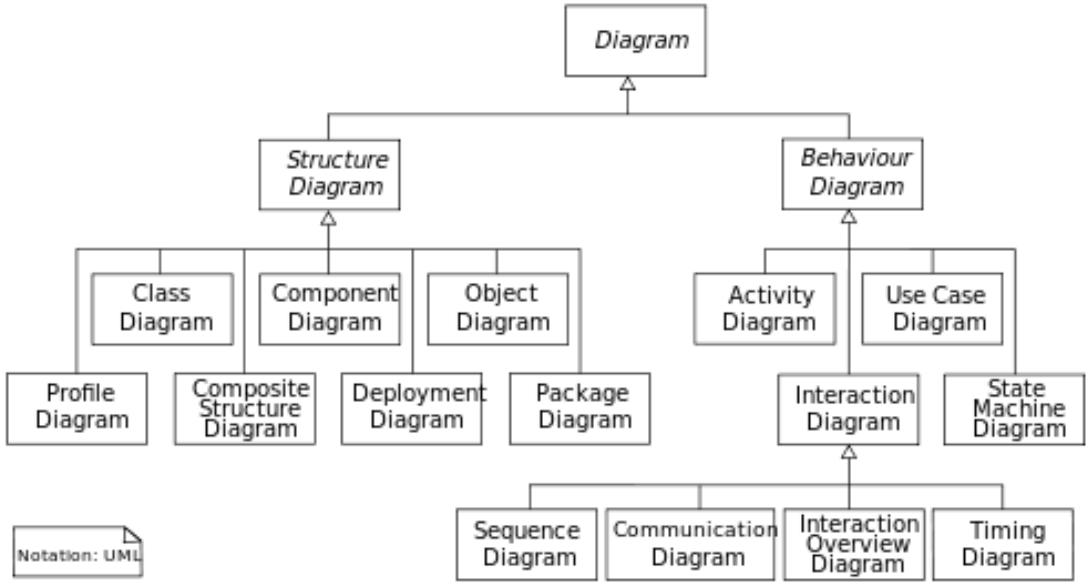
- **Composition** is used to define complex constraints from simple equations

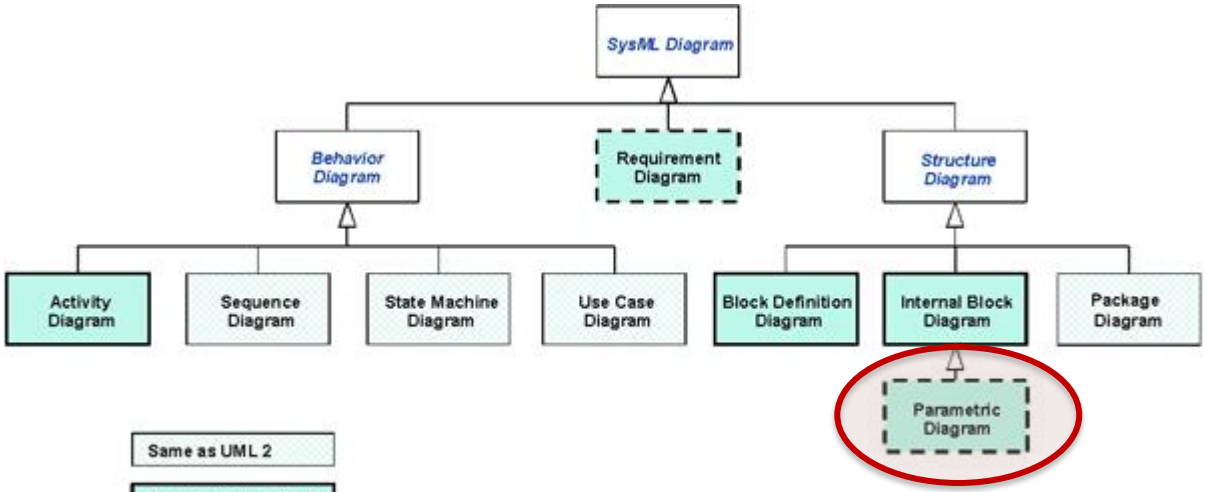


Parametric diagram

Specification of bindings between system parameters

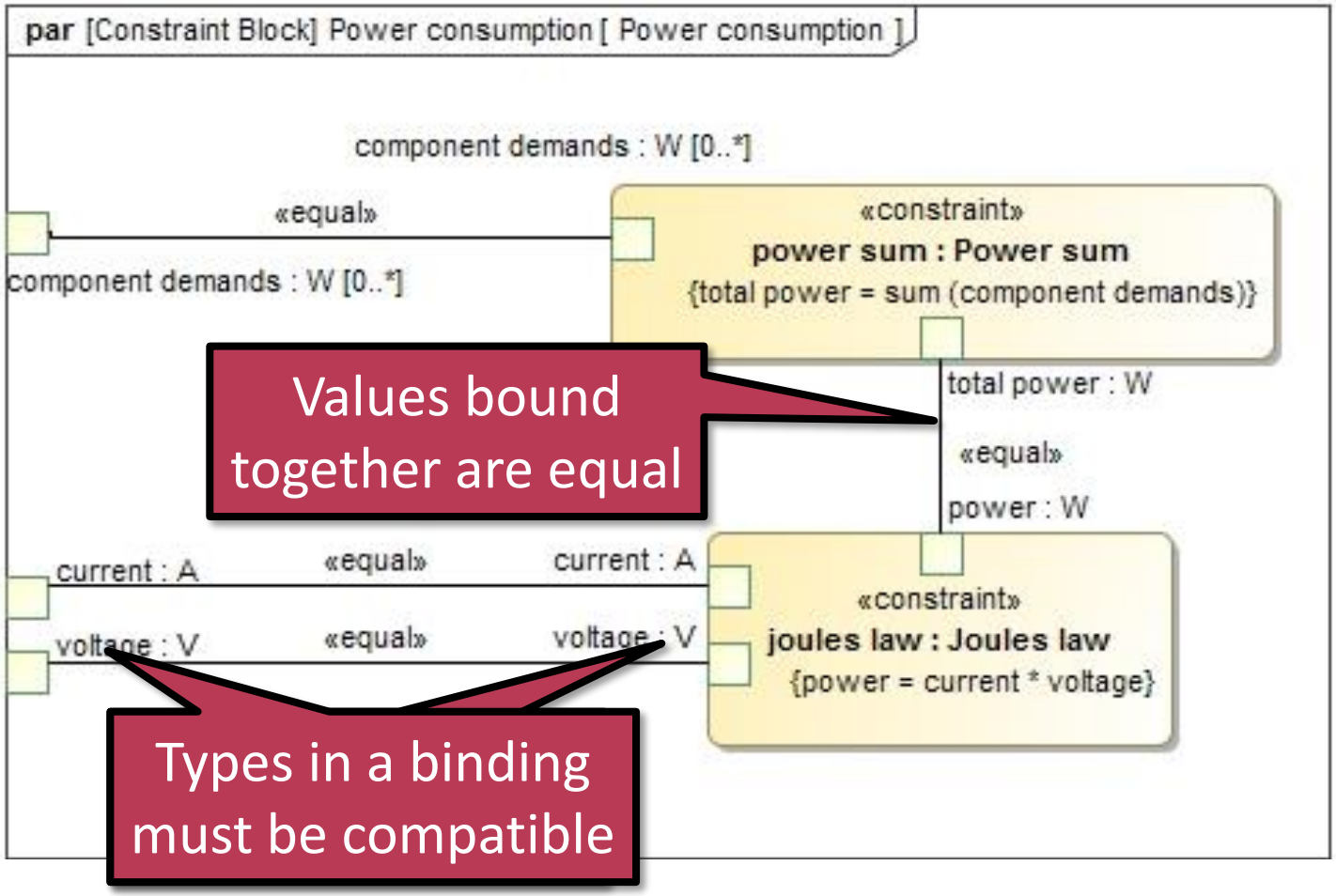
Parametric Diagram (PAR)





Parameter bindings

- Goal: describe the application of constraints in a particular context



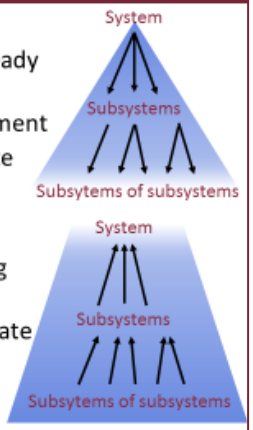
Applications of parametrics

- Parametric specification
  - Define parametric relationships in the system structure
- Parametric analysis
  - Evaluating constraints on the system parameters to calculate values and margins for a given context
  - Checking design alternatives
  - Tool support: ParaMagic plug-in for MagicDraw
- There are modeling standards with better support for this modeling aspect…
  - …such as Modelica

Summary

Top-down and bottom-up design

- Top-down: using decomposition
  - ☺ When designing a subsystem, its goal is already known
  - ☹ There are no working parts during development
  - ☹ Problems, needs of subsystems revealed late
- Bottom-up: using composition
  - ☺ Subsystems can be tested one-by-one
  - ☺ There are always some working parts during development
  - ☹ Exact roles of the subsystems are revealed late
- (Not only in structural modeling…)
- Meet-in-the-middle approach
- Iterative approaches



7

Ports

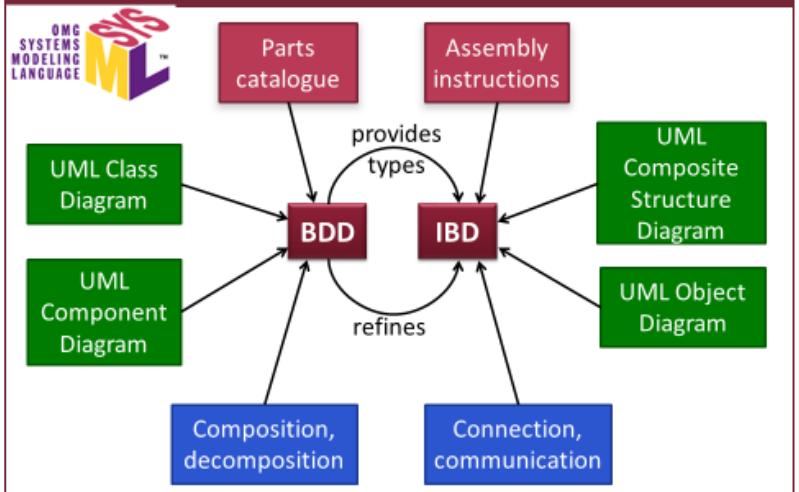
- What is a port?
  - Interaction points with external entities limiting and differentiating the possible connection types

REST API:

Method	URL	Payload	Result
POST	/api/InventoryItem	CreateInventoryItemCommand (input)	Creates a new inventory item
GET	/api/InventoryItem	InventoryItemListDataCollection (output)	Returns all items
PUT	/api/InventoryItem/{id}	RenameInventoryItemCommand (input)	Renames an item

55

Block Definition Diagram vs Internal Block Diagram



30

Types of constraints

- A property of the **system**
- Must hold at any point in time
- "If you realize this system, we assume … (you must guarantee it, or must be a law of nature)"

- System Constraints
  - CPU should receive 12V +- 1V electricity
  - Dissipated heat equals current times voltage

- A property of the **model**
- Can be **enforced** by the editor
- "If a model uses this element, it must …"

- Well-formedness Constraints / Design Rules
  - "if a «Device» Block has an «Electrical» Connector, it must also own a «Ground» Port"

18