AARHUS
UNIVERSITET

26. FEBRUARY 2014

TISYE1 – Lecture 5

Model-based Systems Engineering with SysML

SUNE WOLFF (SWO@ENG.AU.DK)
INDUSTRIAL POSTDOCTORAL RESEARCHER

About the lecturer

- Sune Wolff, Industrial PhD
- Industrial Postdoctoral researcher at Department of Engineering, AU
- Research area:
  - Modelling and simulation of embedded systems
  - Cross-disciplinary collaboration
  - 3D visualisation of complex systems
- SysML experience
  - Used extensively during PhD studies for decomposing complex multi-disciplinary systems
  - Taught courses in SysML
    - PhD course at DTU
    - MasterClass in Systems Engineering for engineers from industry
    - 2nd semester Systems Engineering

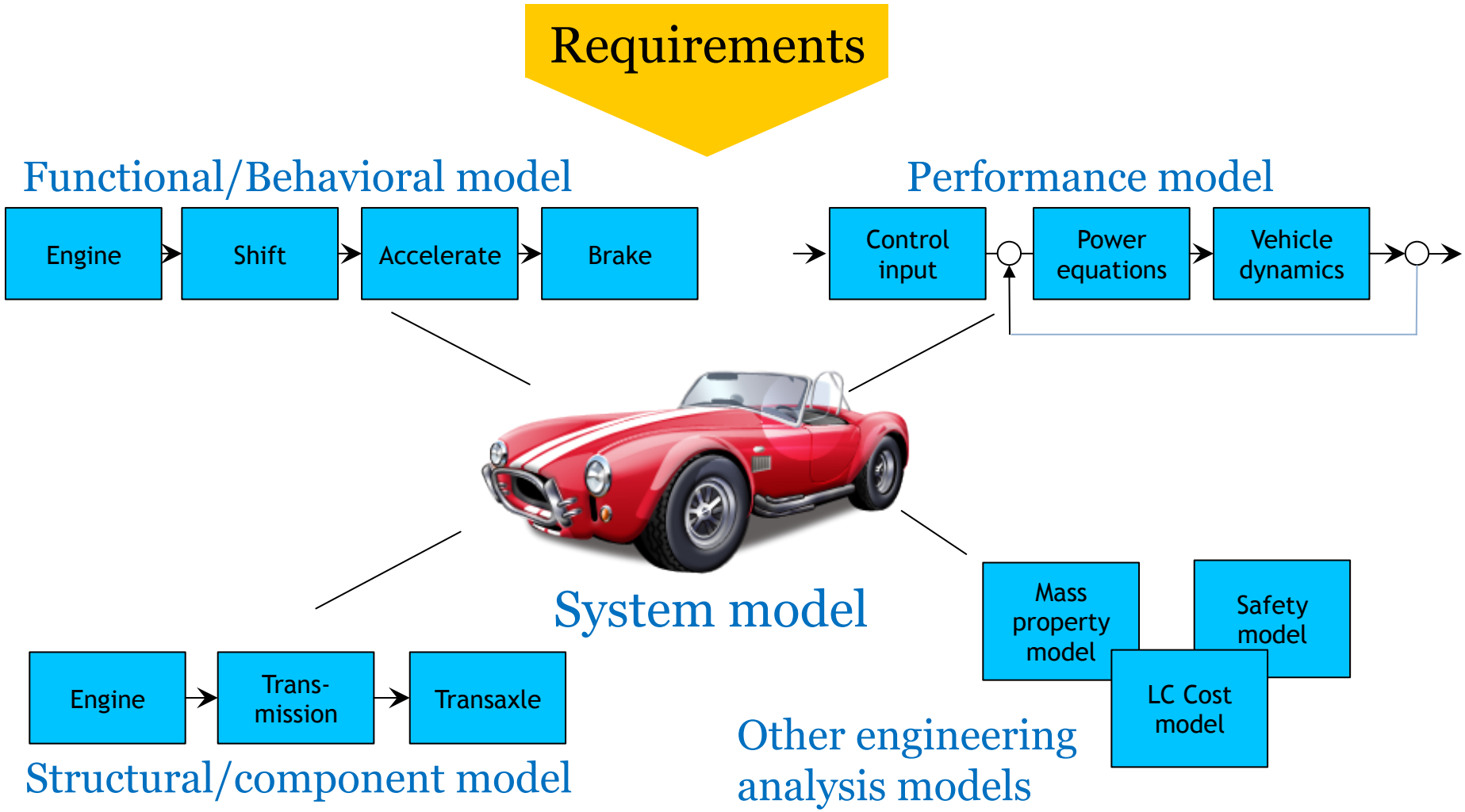
Plan for the Lecture

- Introduction to MBSE
- What is SysML?
- New SysML concepts and diagrams
  - SysML Requirements
  - SysML Structure
  - SysML Behavior
  - SysML Parametric
- SysML Tools
- Perspectives for SysML

Model-based systems engineering (MBSE)

- From **document-oriented** to **model-based** approach.
- A model-based approach requires **modeling concepts and tools.**
- MBSE: producing and controlling a coherent **system model** as opposed to a coherent set of documents.
- SysML is created to realize an MBSE approach based on a system model of the wanted system.
- SysML is **a modeling language** not a System Engineering (SE) process.

Systems modeling – perspectives



Methodologies

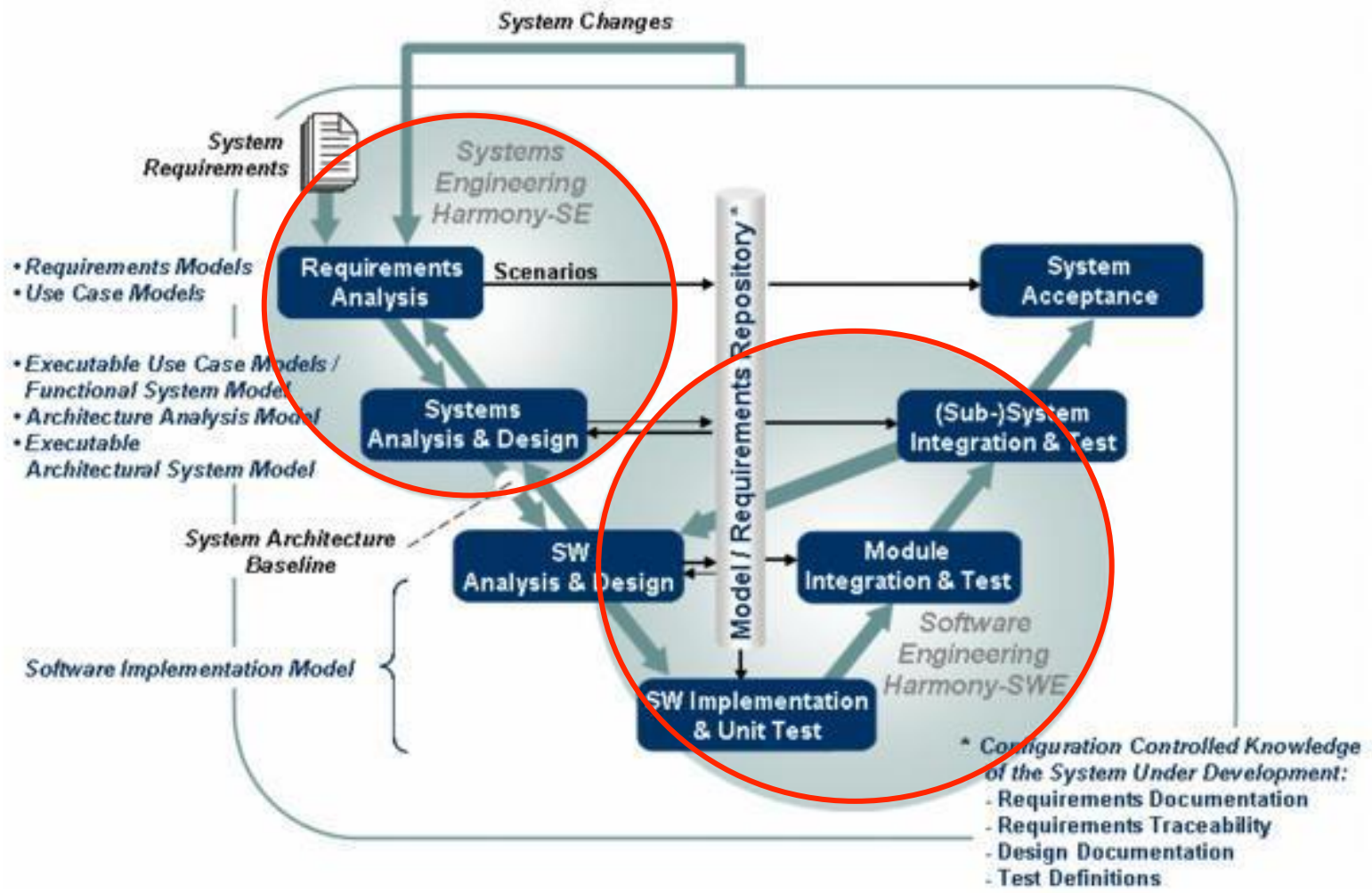
... recognized by INCOSE

- IBM's Harmony SE
- INCOSE's Object-Oriented Systems Engineering Method (OOSEM)
- IBM's Rational Unified Process for systems Engineering (RUP-SE)
- Vitech's MBSE Methodology and SDL
- Jet Propulsion Laboratory's (JPL) state analysis (SA)

The Harmony process

- The Harmony process facilitates a seamless transition from **Systems Engineering** to **Software Engineering.**
  - It uses SysML exclusively for system representation and specification
- Harmony process characteristics:
  - a scenario-driven and iterative development process
  - promotes reuse of test scenarios throughout system development

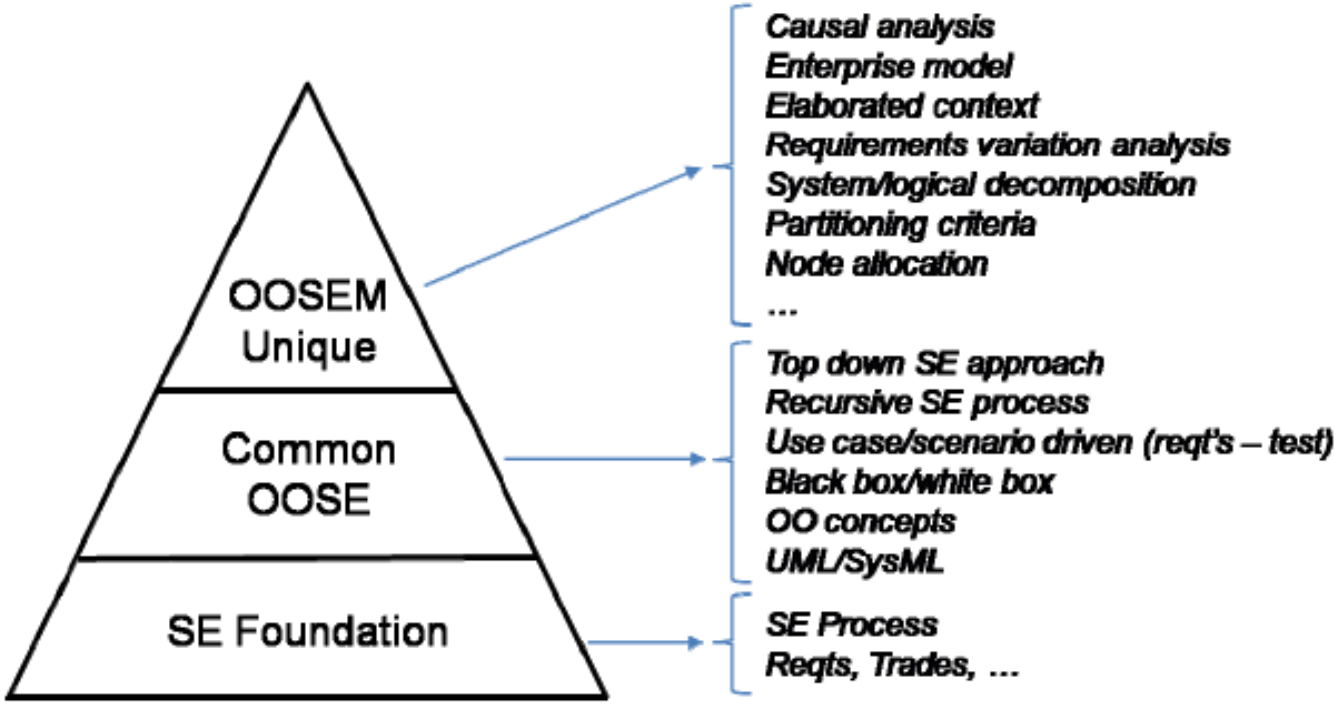
Harmony process for Systems Engineering



Benefits of the Harmony process

- The Harmony process allows systems engineers to **find design errors** early in the development.
- **Customer requests** can be more efficiently assessed, incorporated, and given timely feedback.
- The greatest benefit of a model-driven process is, however, **improved communication**
  - between engineering disciplines.
  - between technical and non-technical parties.
  - using different levels of abstraction.
  - avoids information overload.

Object-Oriented SE Method (OOSEM)



OOSEM is a hybrid approach that leverages object oriented techniques and an SE foundation

What is SysML?

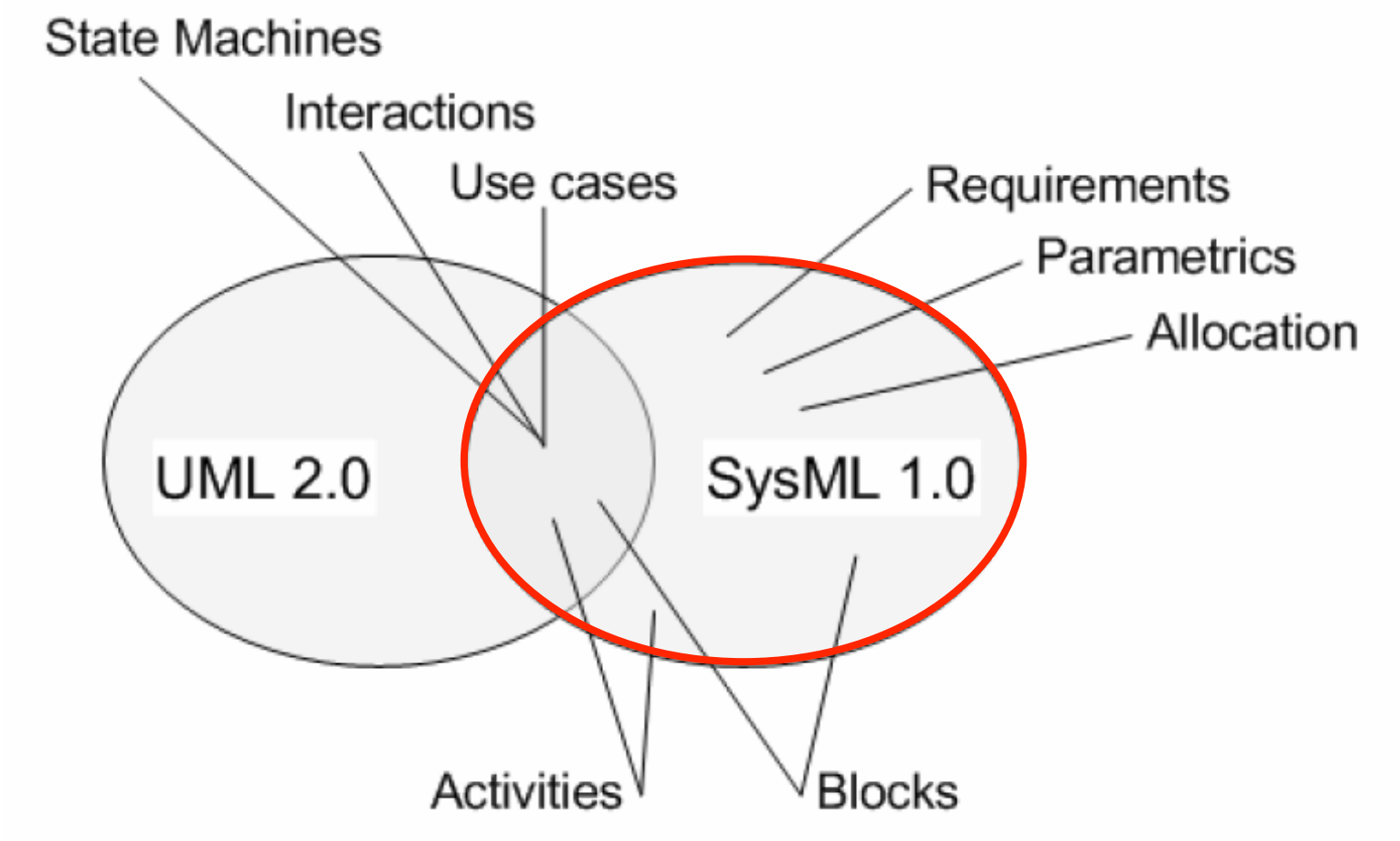
- A graphical modeling language created in response to the UML for Systems Engineering RFP developed by **OMG** and **INCOSE**.
  - a UML Profile that represents a **subset of UML 2** with important extensions
- Supports the specification, analysis, design, verification and validation of systems that include **hardware**, **software**, **data**, **personnel**, **procedures**, and **facilities**.
- Supports model and data interchange via **XMI**.

SysML is a critical enabler for model-driven or model-based systems engineering

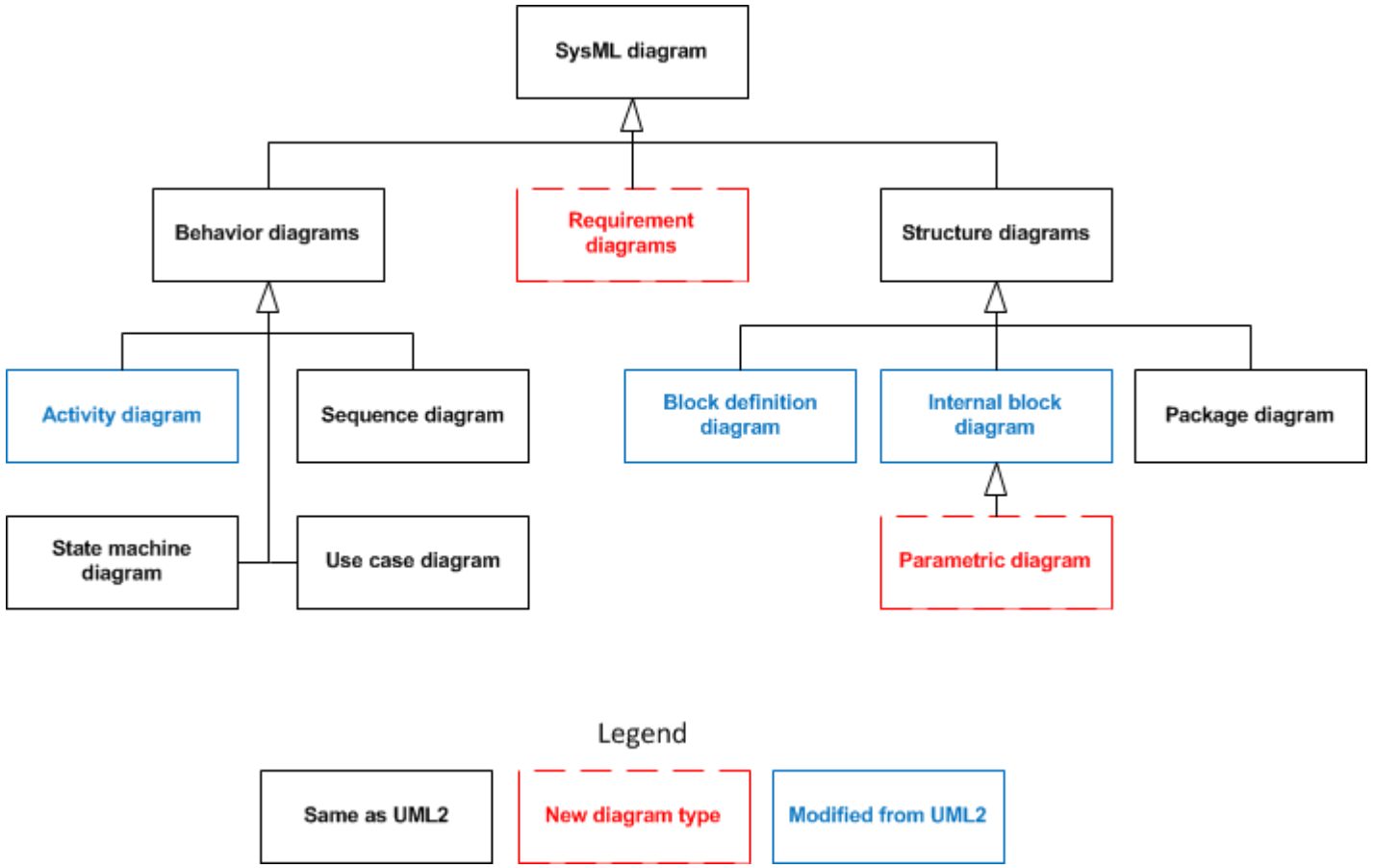
SysML specification history

- November 1997: UML V1.1 launched by OMG
- March 2003: The UML for Systems Engineering RFP (Request for Proposal) was developed jointly by OMG and INCOSE.
  - The SysML specification was developed in response to these requirements by adiverse group of tool vendors, end users, academia, and government representatives.
- September 2007: OMG SysML v.1.0
- November 2008: OMG SysML v1.1 (256 pages)
- June 2010: OMG SysML v.1.2 (260 pages)
- June 2012: OMG SysML v.1.3 (272 pages)
  - www.omg.org/spec/SysML/1.3/

Comparison of SysML and UML



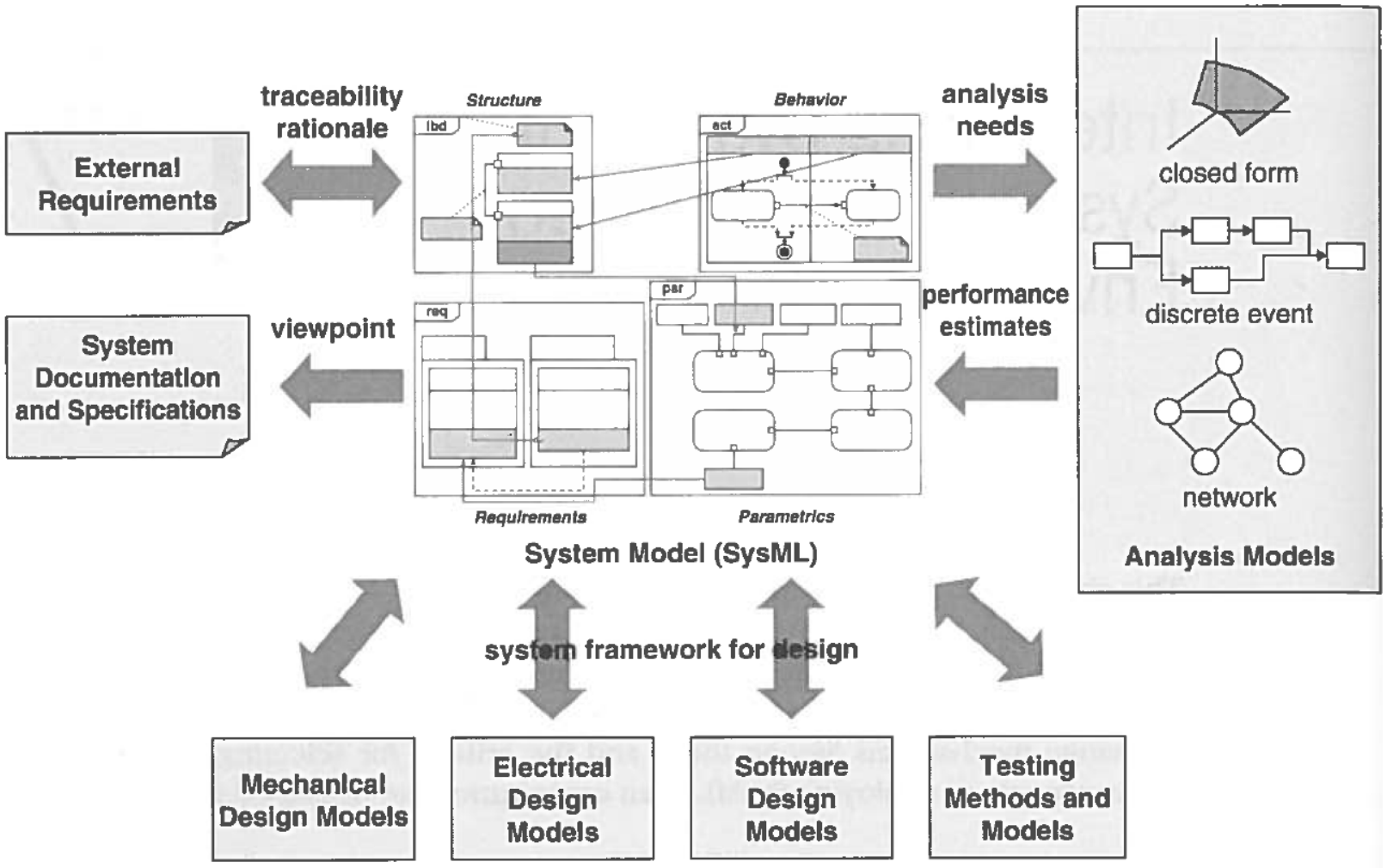
SysML diagram types taxonomy



Major Extensions compared to UML 2.x

- New Diagram Types
  - Requirement Diagram (req)
  - Parametric Diagram (par)
- Structure Diagrams
  - Block Definition Diagram (bdd)
  - Internal Block Diagrams (ibd)
- Activity Diagrams
  - extensions for **continuous flow** modeling
  - extensions to support control operators

System model for analysis and traceability

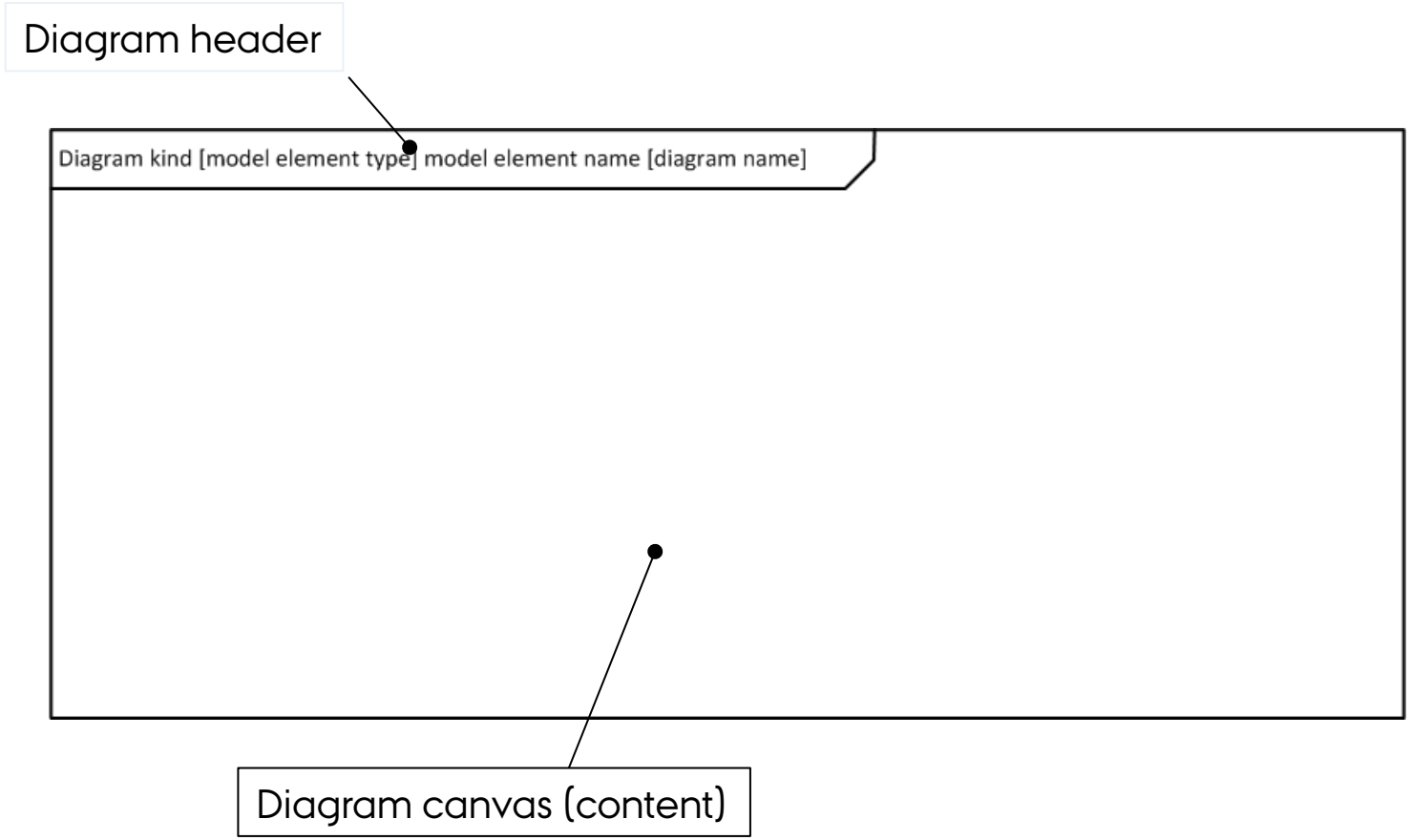


Project activities using SysML

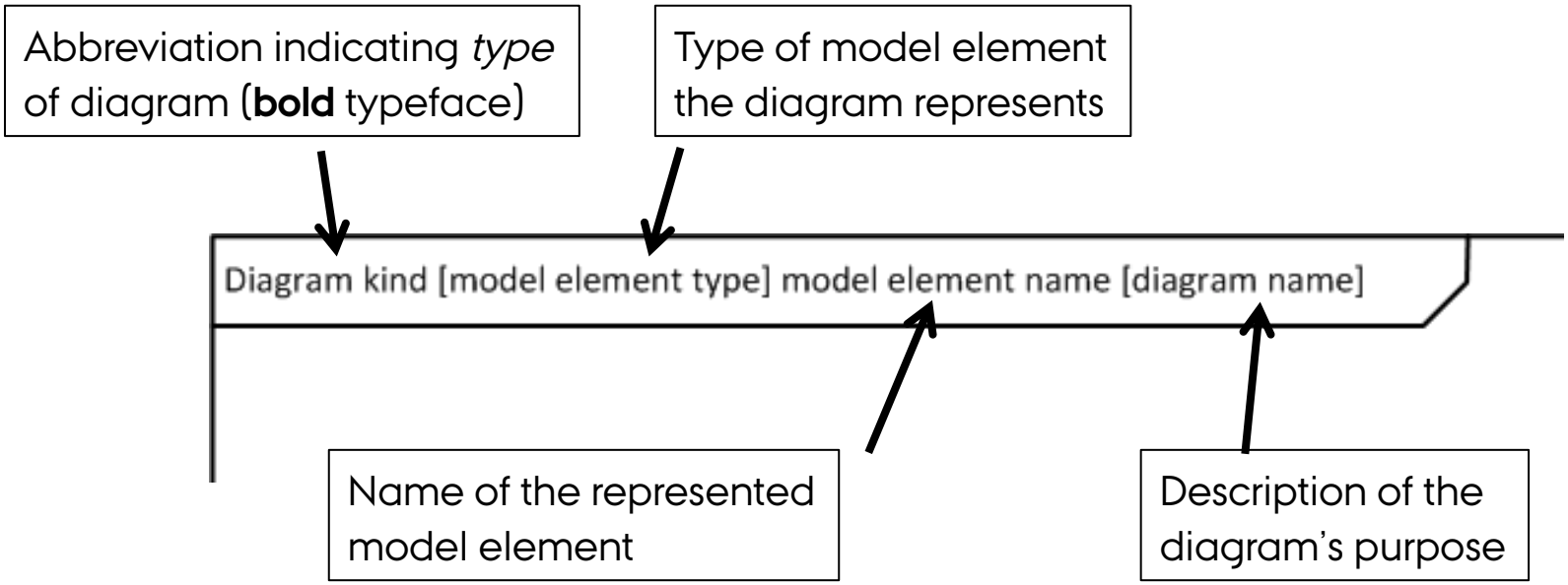
1. Capture and analyze black box system requirements
   - System Context & System Use Cases, Requirement diagrams.
2. Develop one or more candidate system architectures
   - Block Definition & Internal Block diagrams.
3. Perform engineering trade-off analysis to evaluate and select the optimal architecture
   - Parametric diagrams
4. Specify component requirements and their traceability to system requirements
   - Requirement diagram
5. Verify the system design by executing system-level test cases
   - Parametric diagrams

SysML: Diagram frame

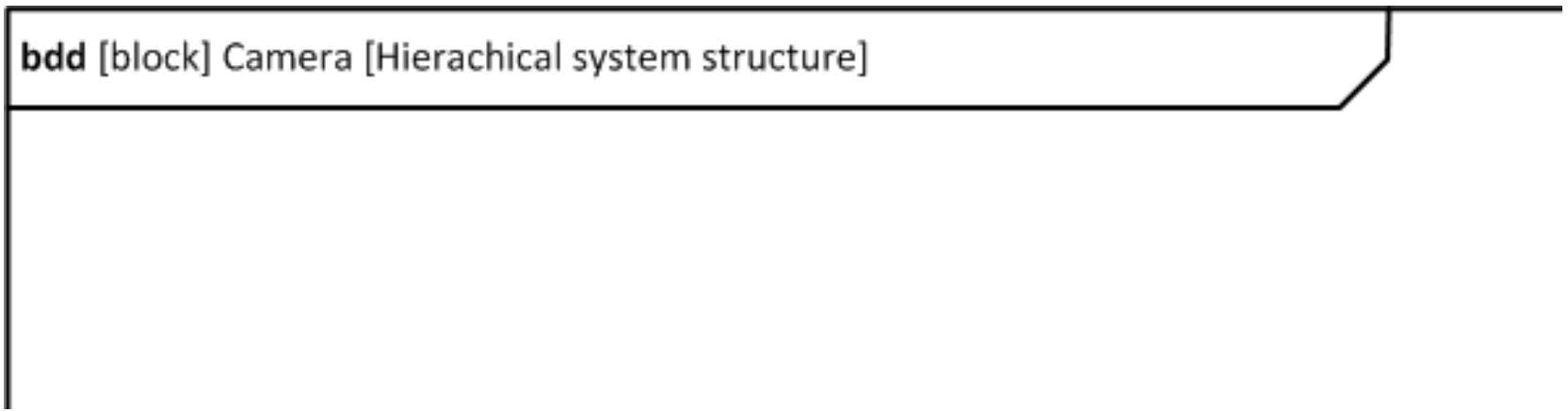
- The diagram frame consists of header and canvas



SysML: Diagram header



SysML: Diagram header - example

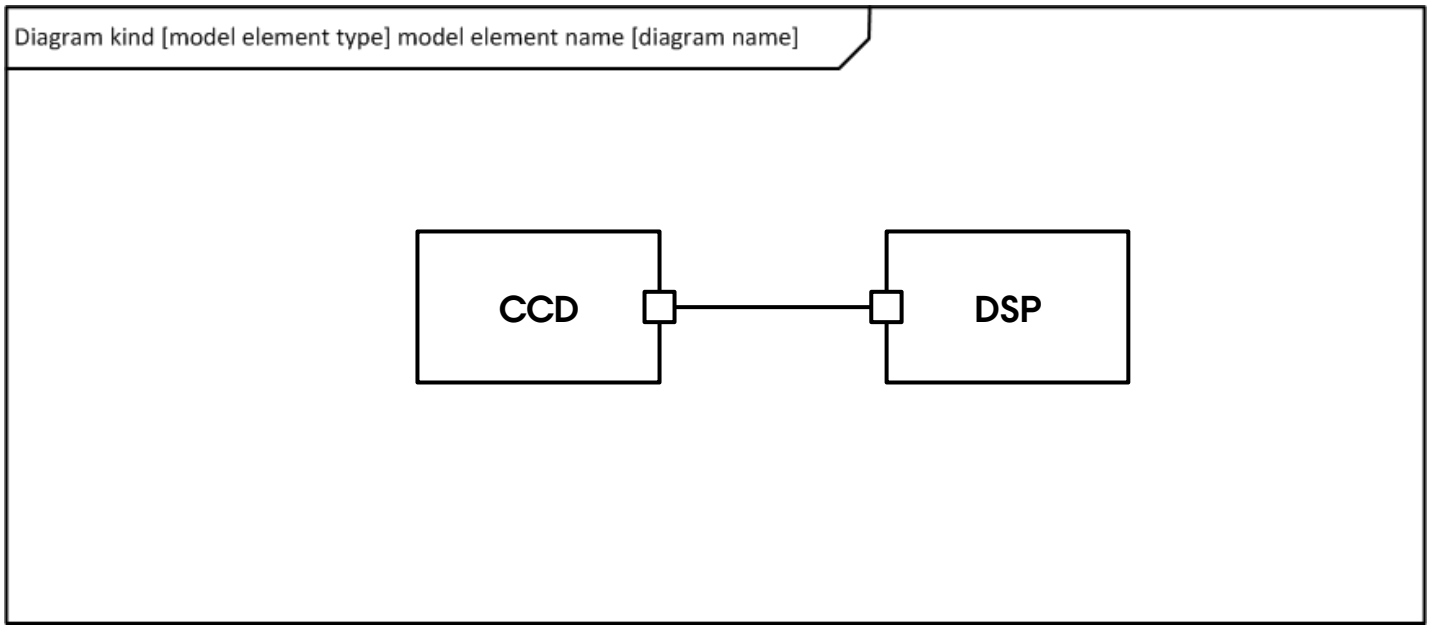


This is a *block definition diagram* (bdd), defining the *hierarchical system structure* of the *block Camera*

Items in brackets are optional - *model element type* frequently omitted, *diagram name* frequently included

SysML: Diagram canvas

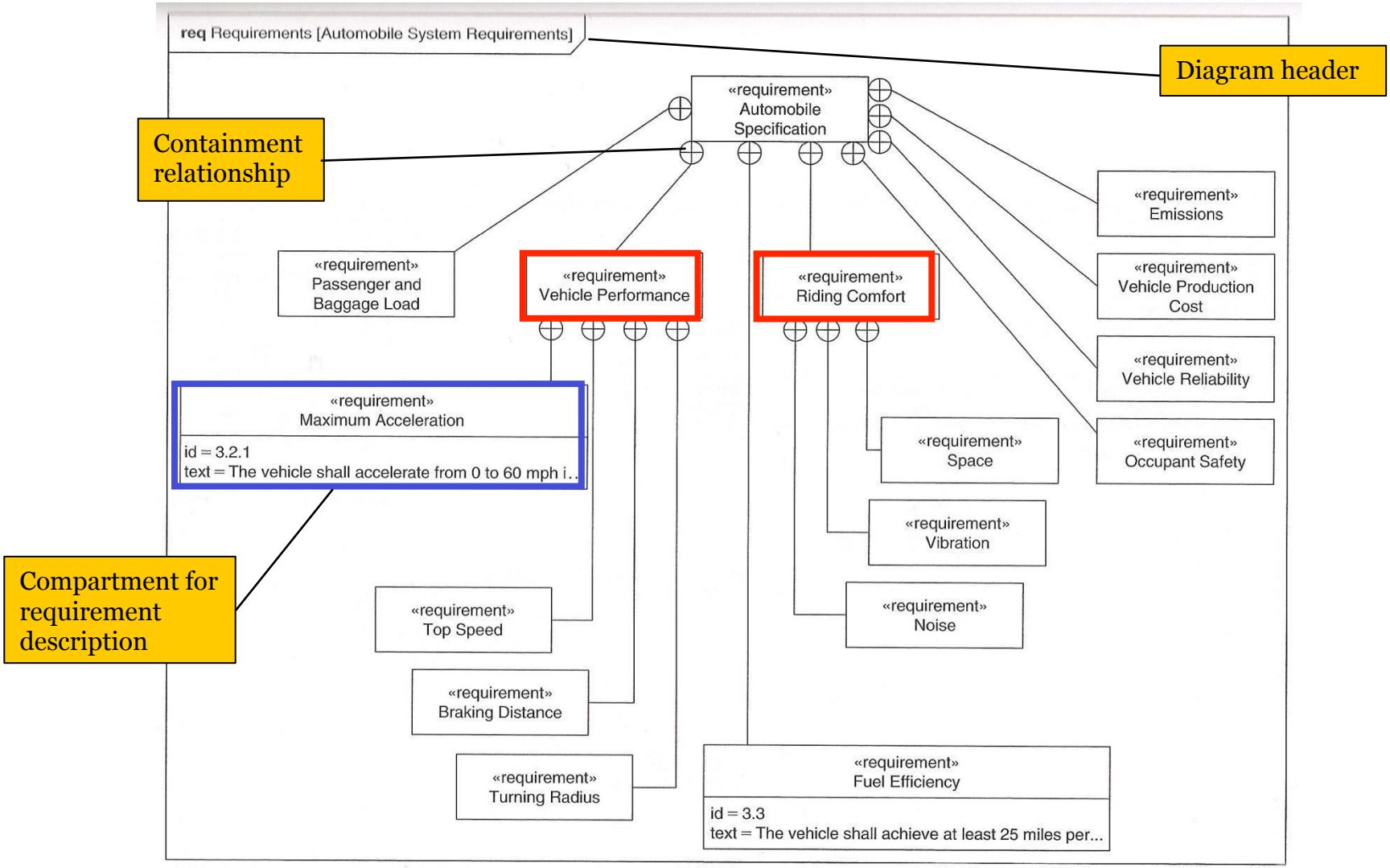
- The diagram canvas holds the modeling elements



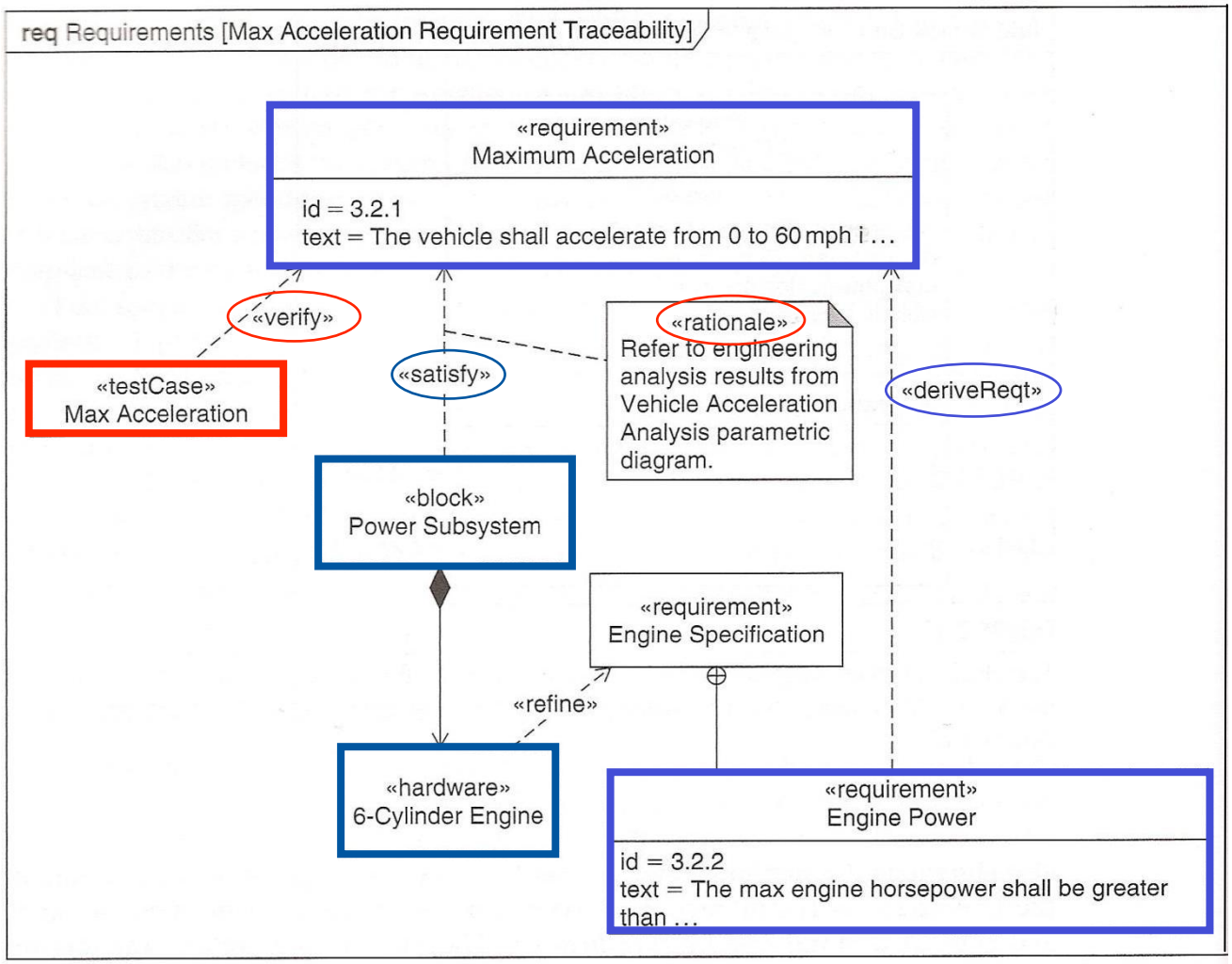
1. SysML Requirements

- Requirement Diagram – NEW diagram type
- Graphical visualization of requirements
  - Functional
  - Non-functional
- Requirements can graphically be related to:
  - Other requirements
  - Design elements
  - Test cases
- Standard stereotypes:
  - **derive**, **satisfy**, **verify**, **refine**, **trace** and **copy**
  - Used for requirement **traceability**

Requirements diagram (req)



Requirements traceability

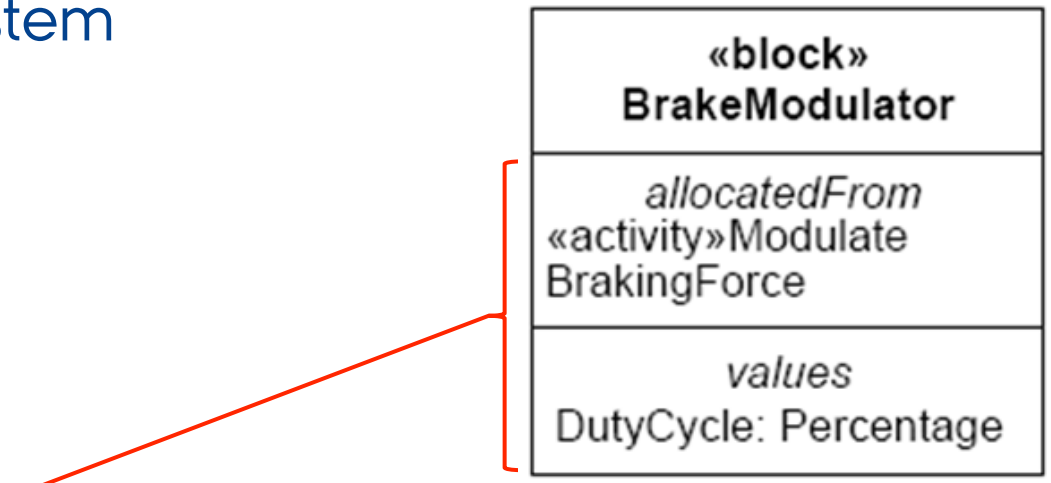


2. SysML structure diagrams

- UMLs class concept is replaced with the **Block** concept
- A Block connects to other blocks via **Ports**
- Class diagrams are replaced with **Block Definition Diagrams (bdd)**
- Each Block has an **Internal Block Diagram (ibd)** where the internal parts are connected via **ports**
  - a replacement for class composite diagrams
- **Ports** can connect **discrete** as well as **continuous flows of material or information**

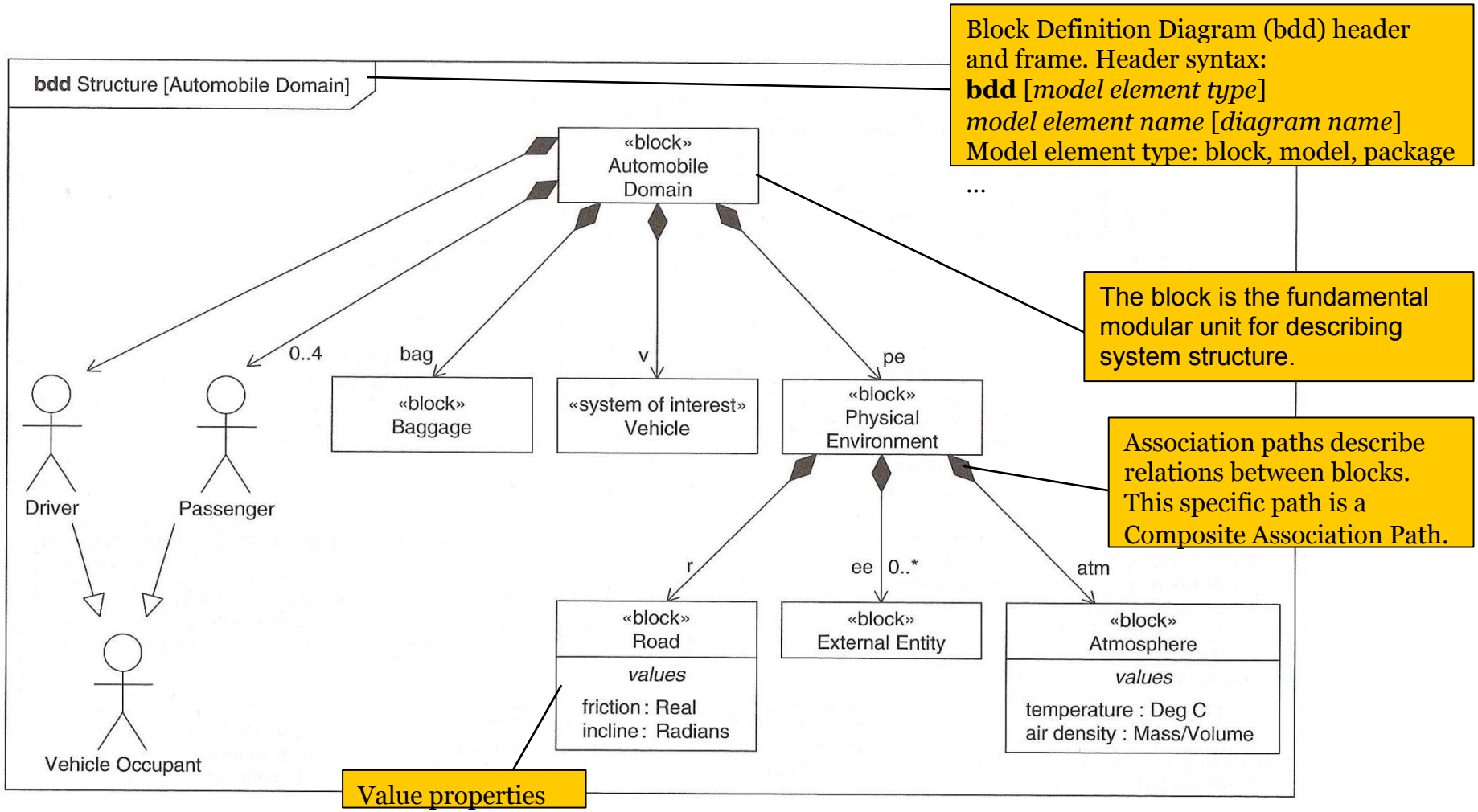
Blocks are basic structural elements

- Provides an unifying concept to described the structure of an element or system
  - Hardware
  - Software
  - Data
  - Procedure
  - Facility
  - Person

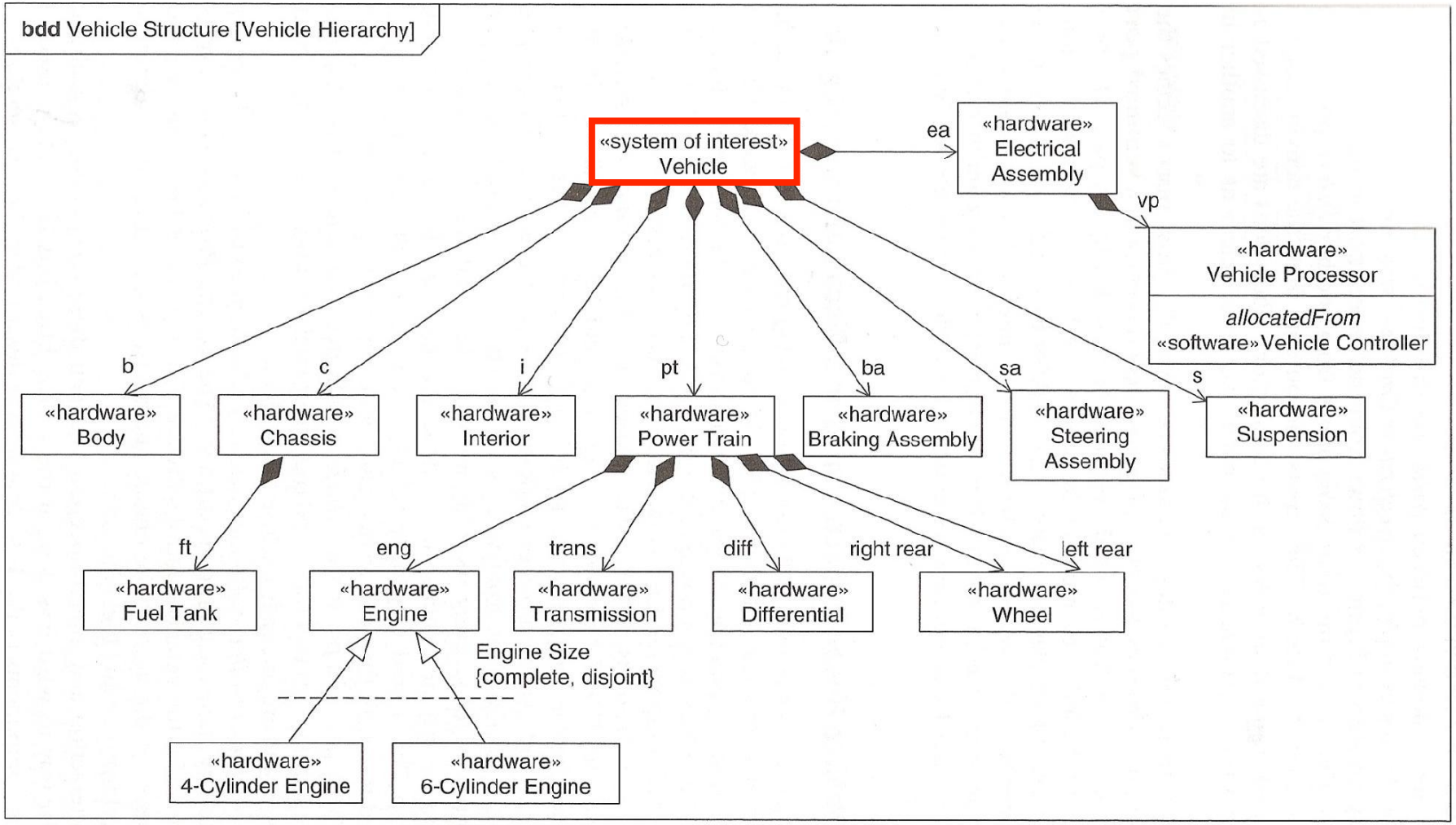


- Multiple compartment can describe the block characteristics
  - Properties (parts, references, values)
  - Operations
  - Constraints
  - Allocations to the block (e.g., activities)
  - Requirements the block satisfies

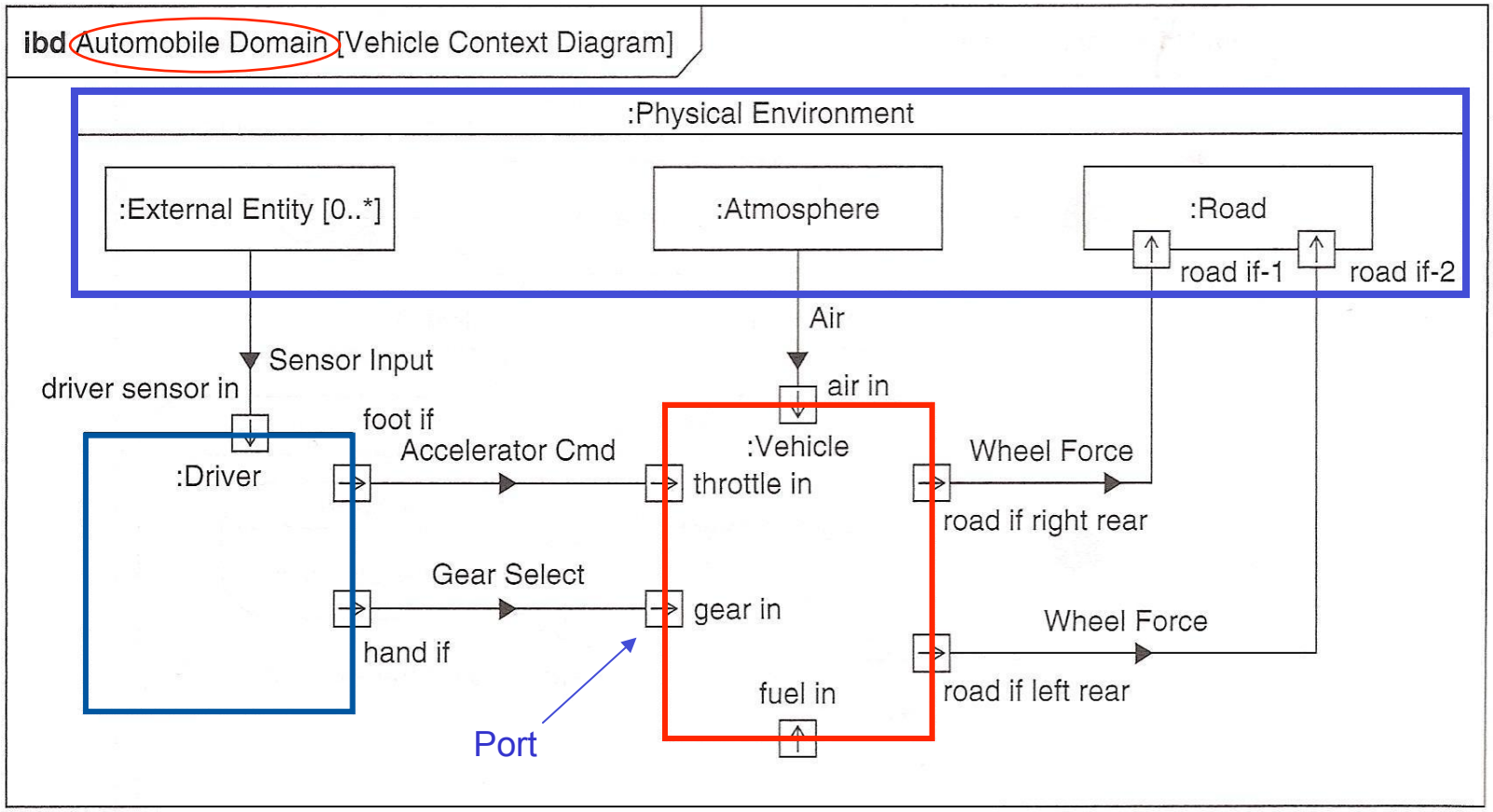
Block definition diagram (bdd)



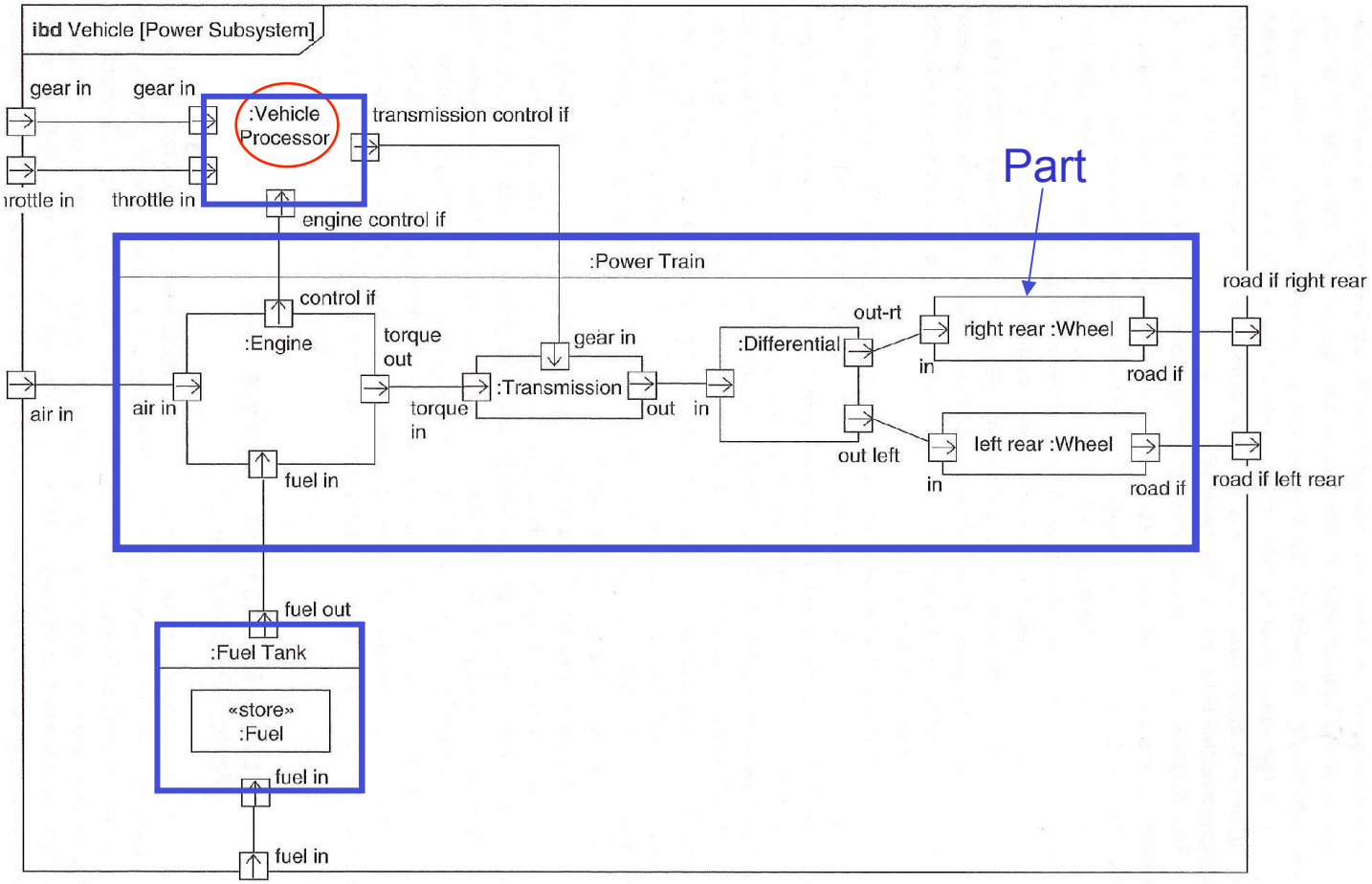
Block definition diagram for vehicle



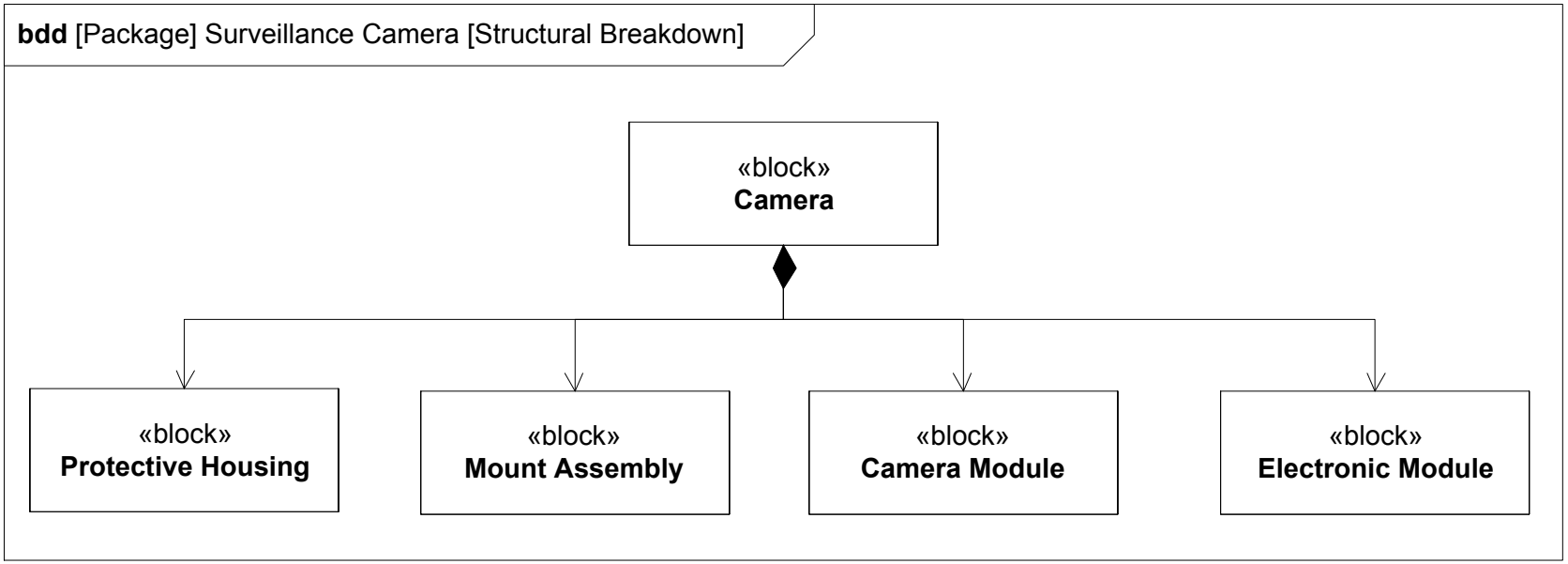
Internal Block Diagram (ibd)



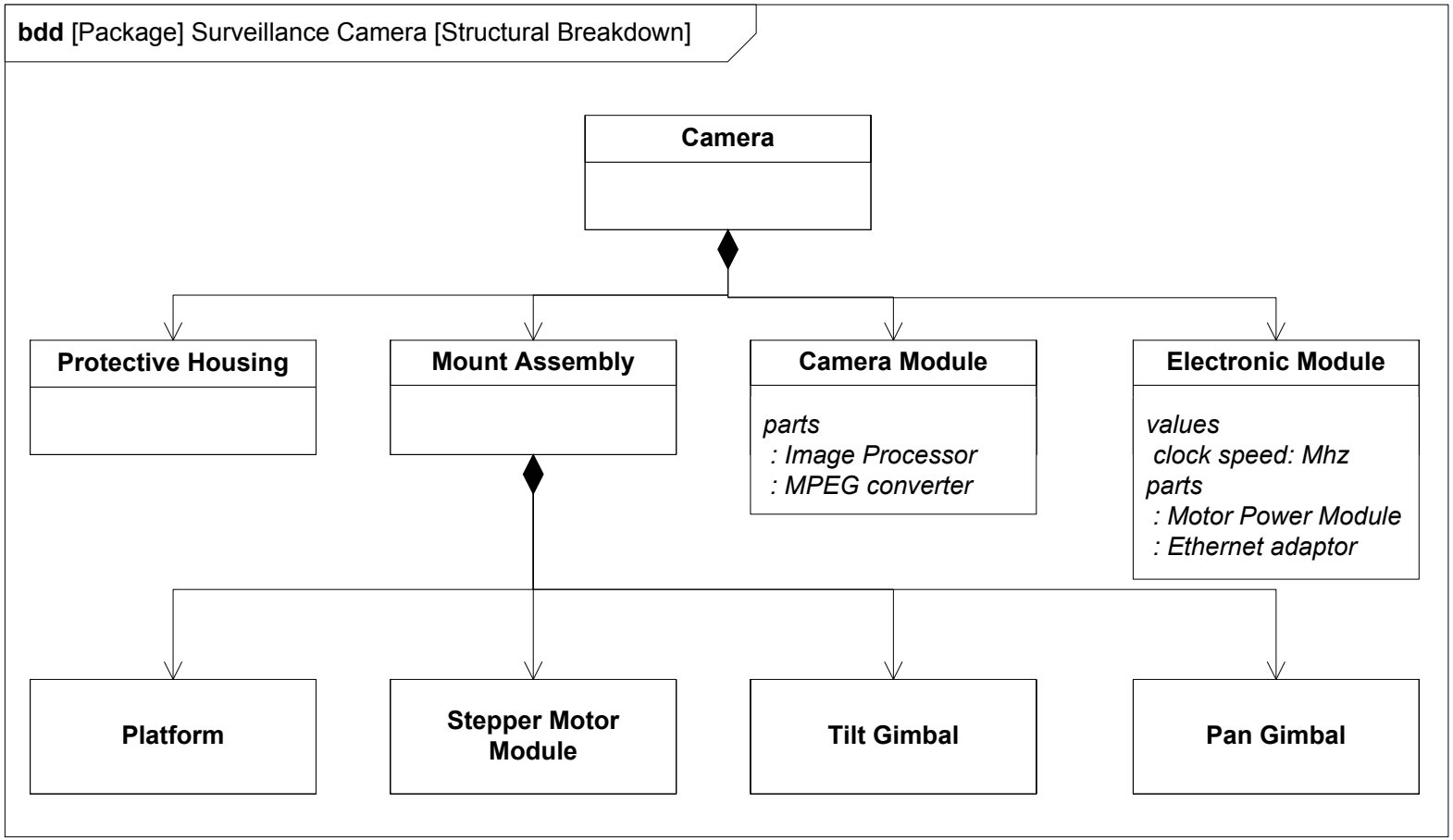
Internal Block Diagram Example



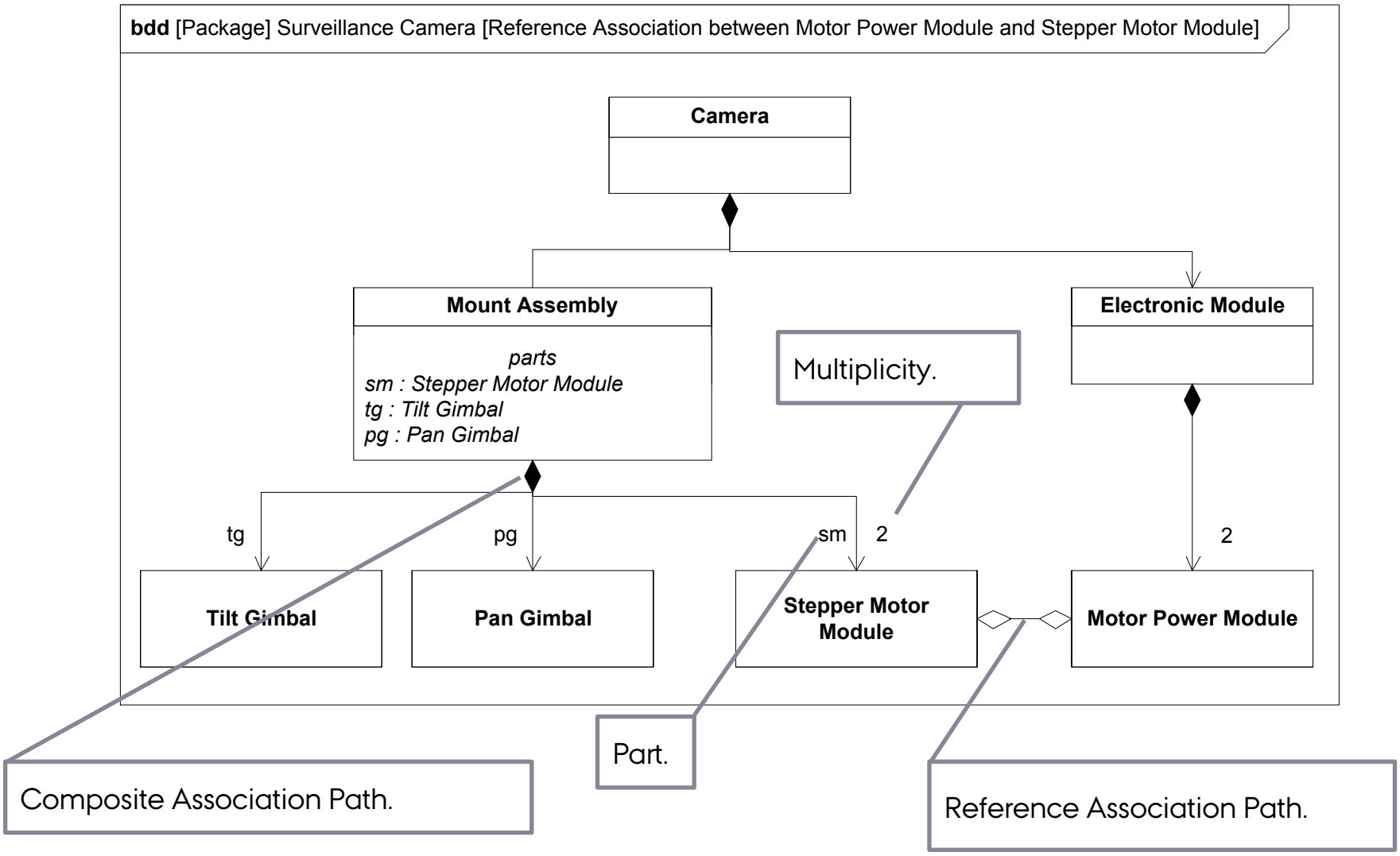
BDD example – surveillance camera



BDD example – surveillance camera

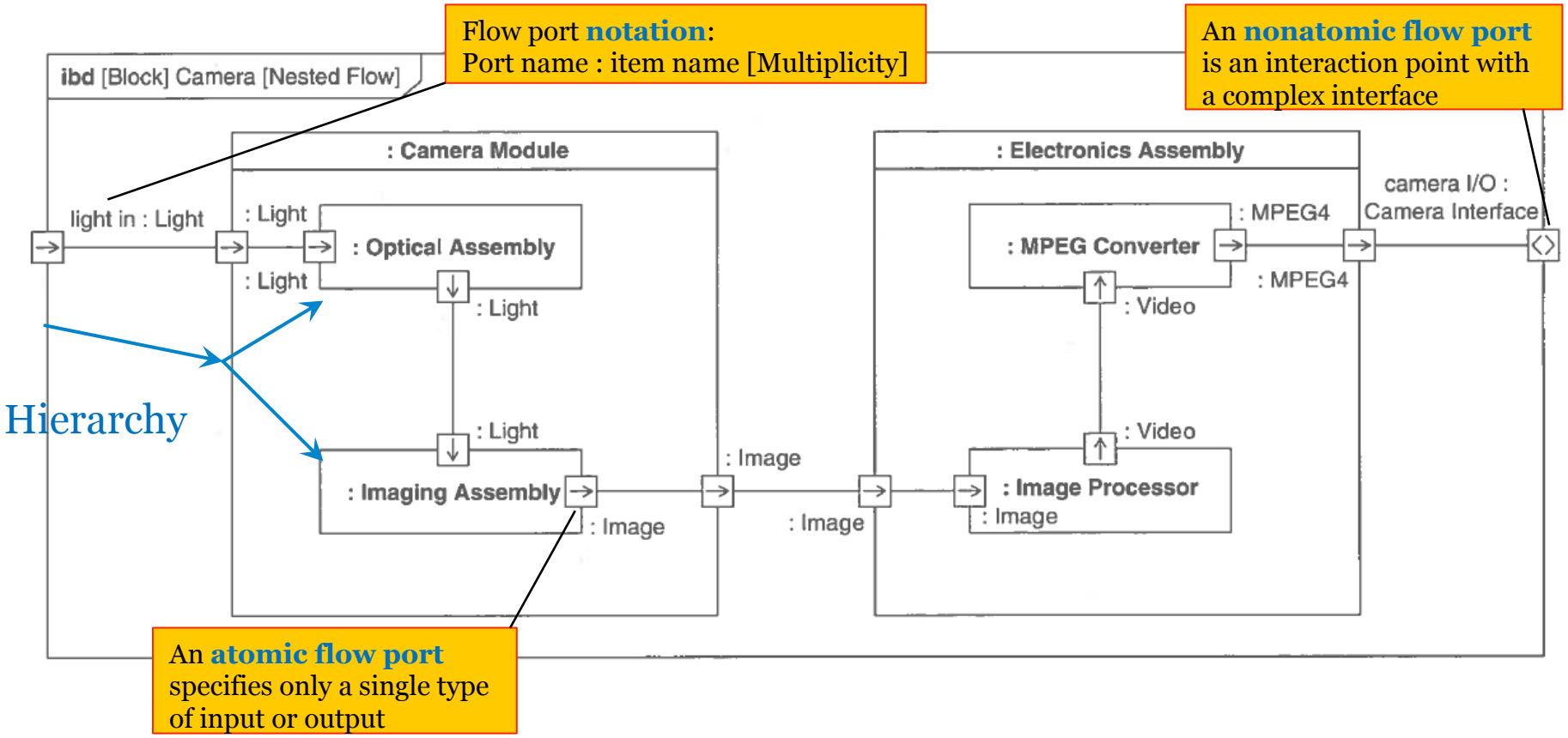


BDD example – surveillance camera

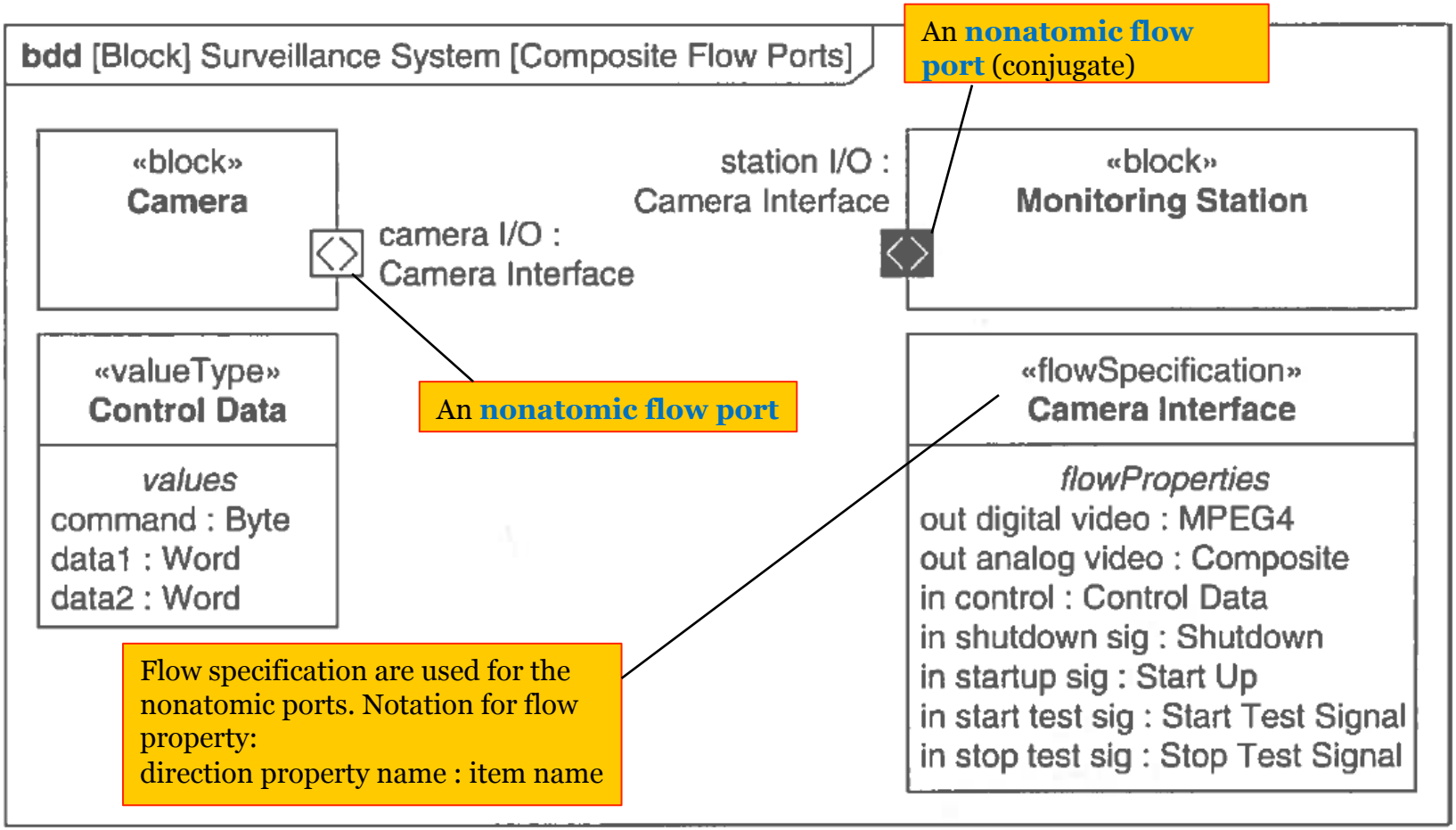


Blocks and atomic flow ports

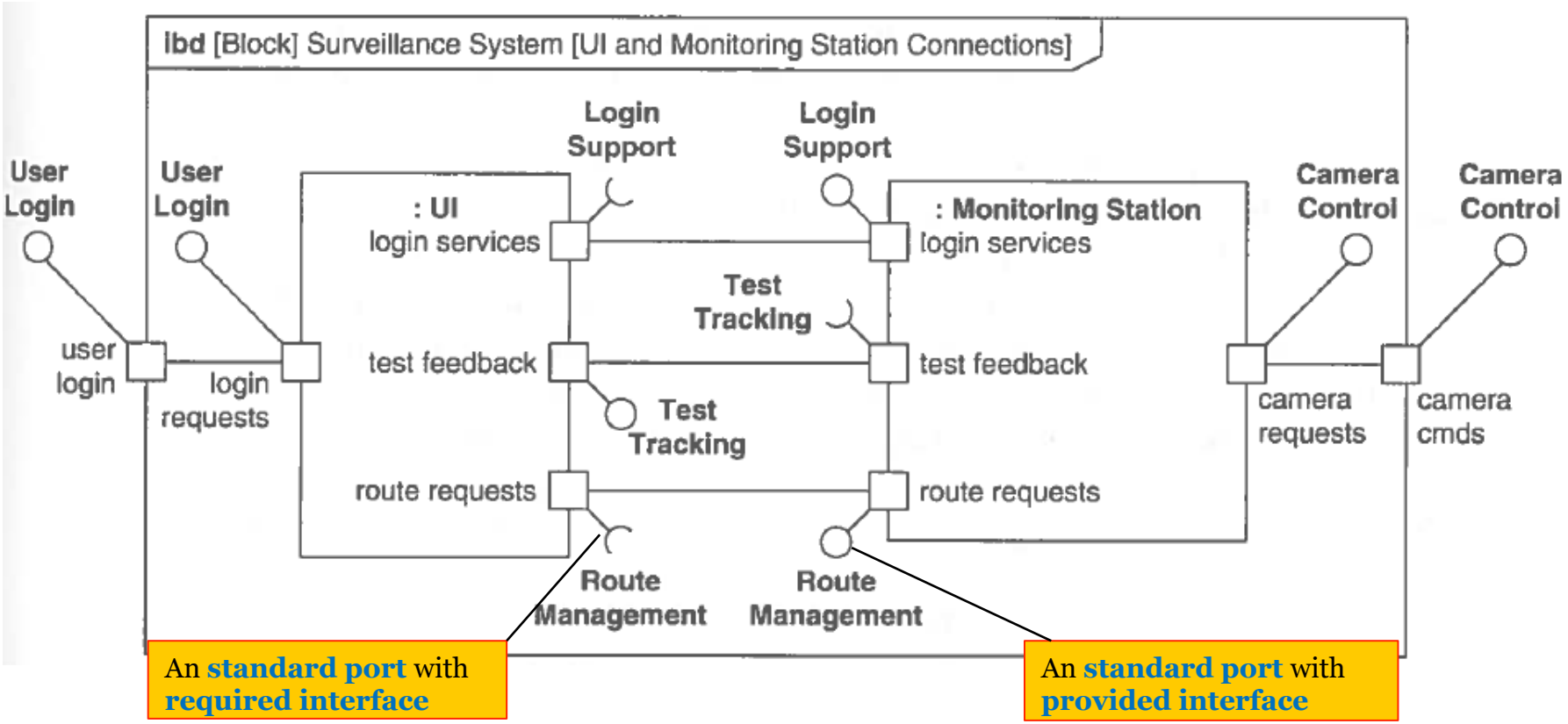
- A **flow port** describes an interaction point for items flowing in or out of a block.



Blocks and flow specifications

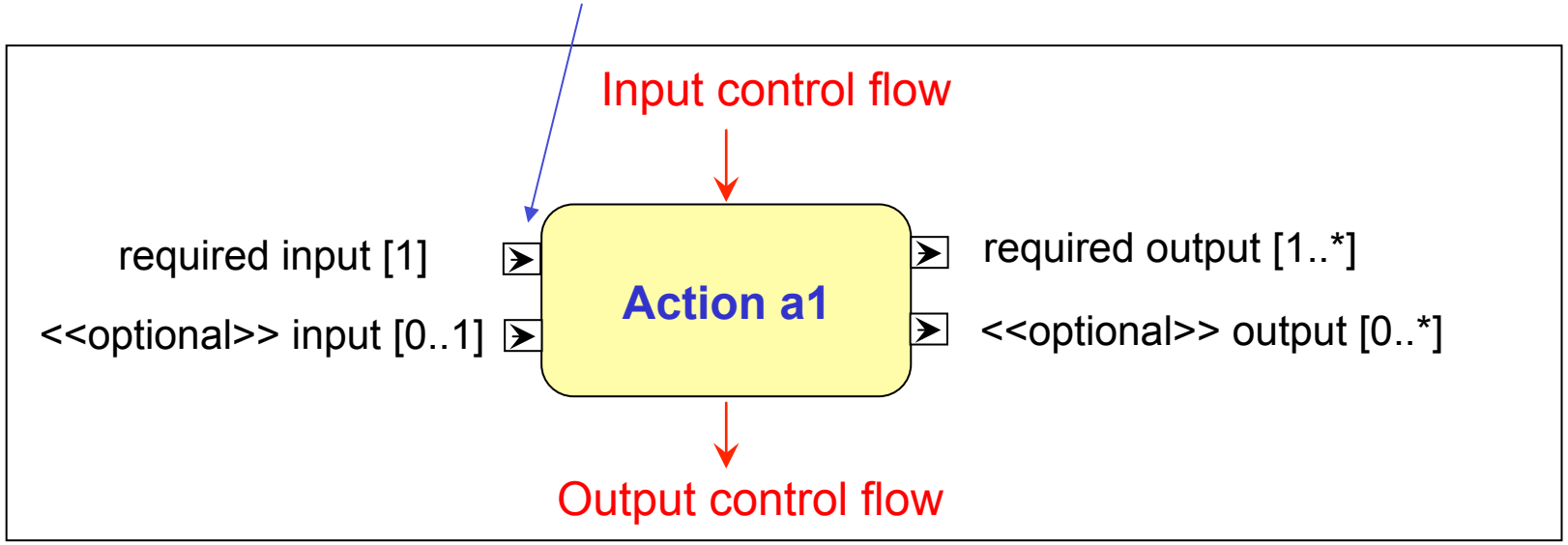


Standard (service based) ports

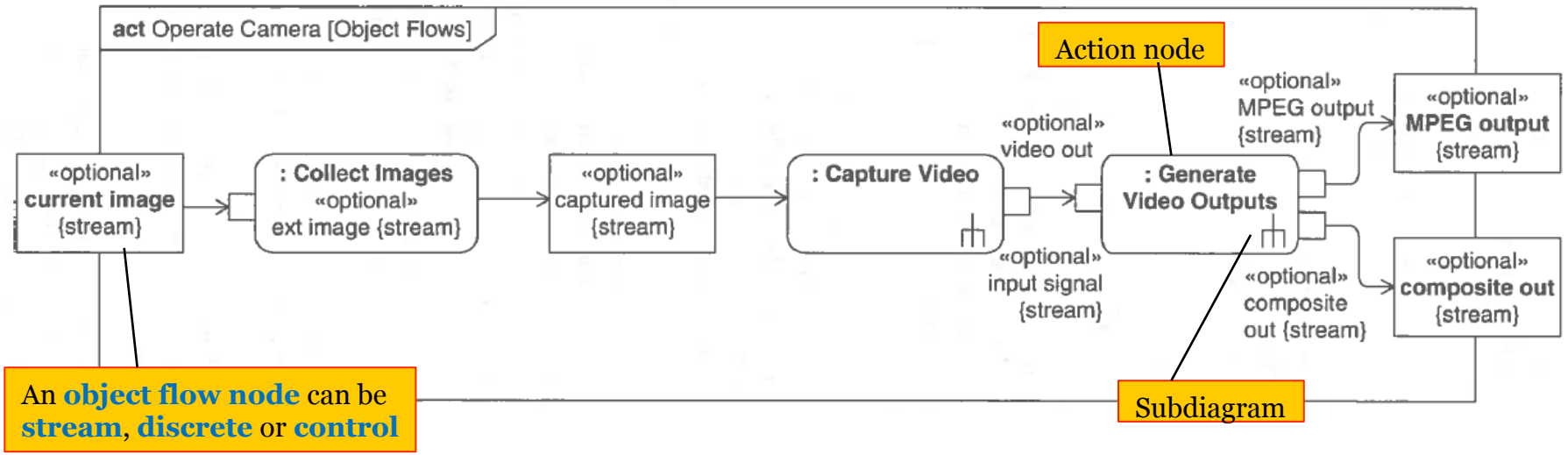


3. SysML behavior

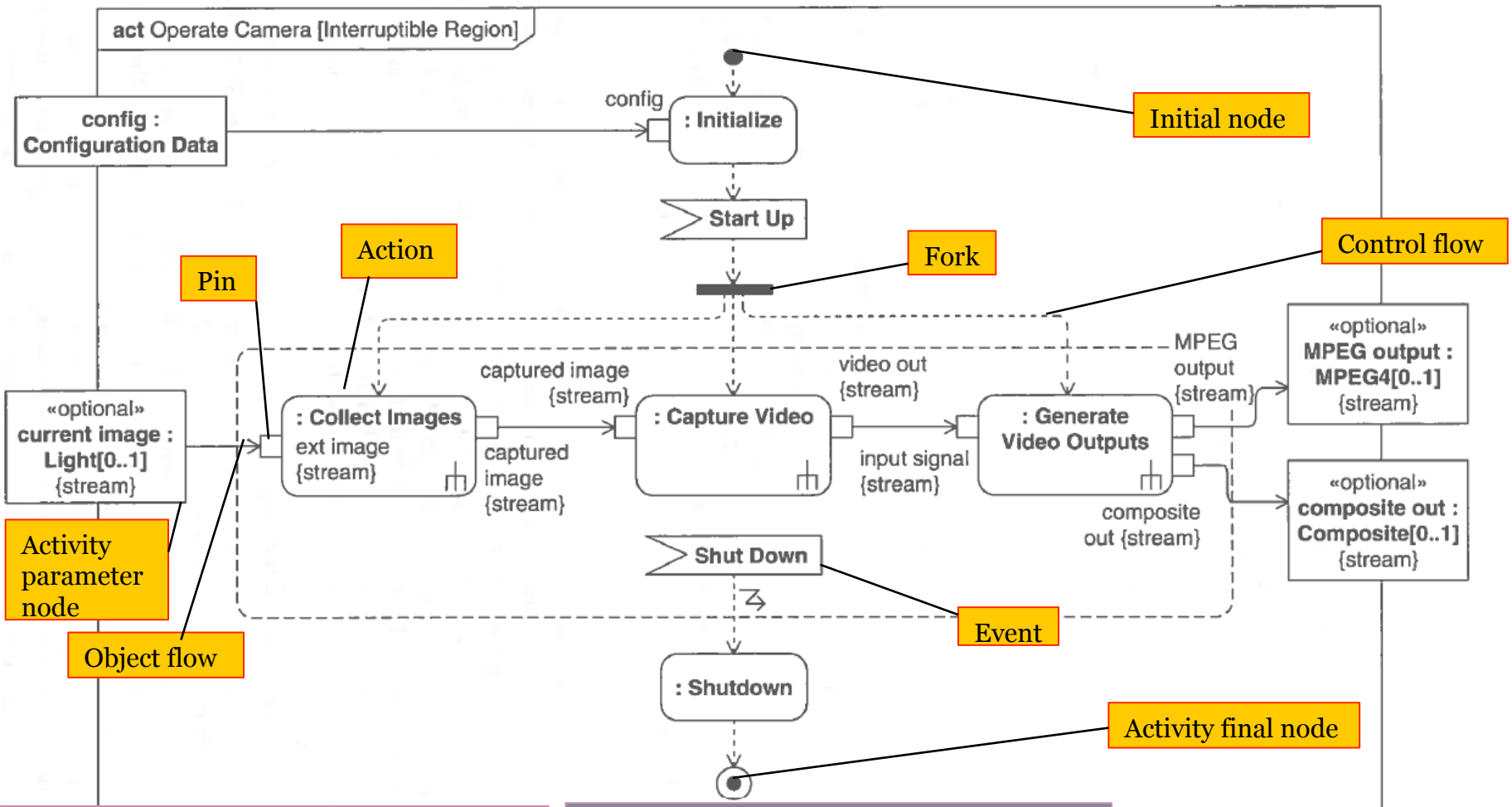
- SysML activities are based on token-flow semantics.
- Tokens corresponds to values of inputs, outputs and control.
- Control flows are introduced.
- Activities can have pins (acting as a buffer).



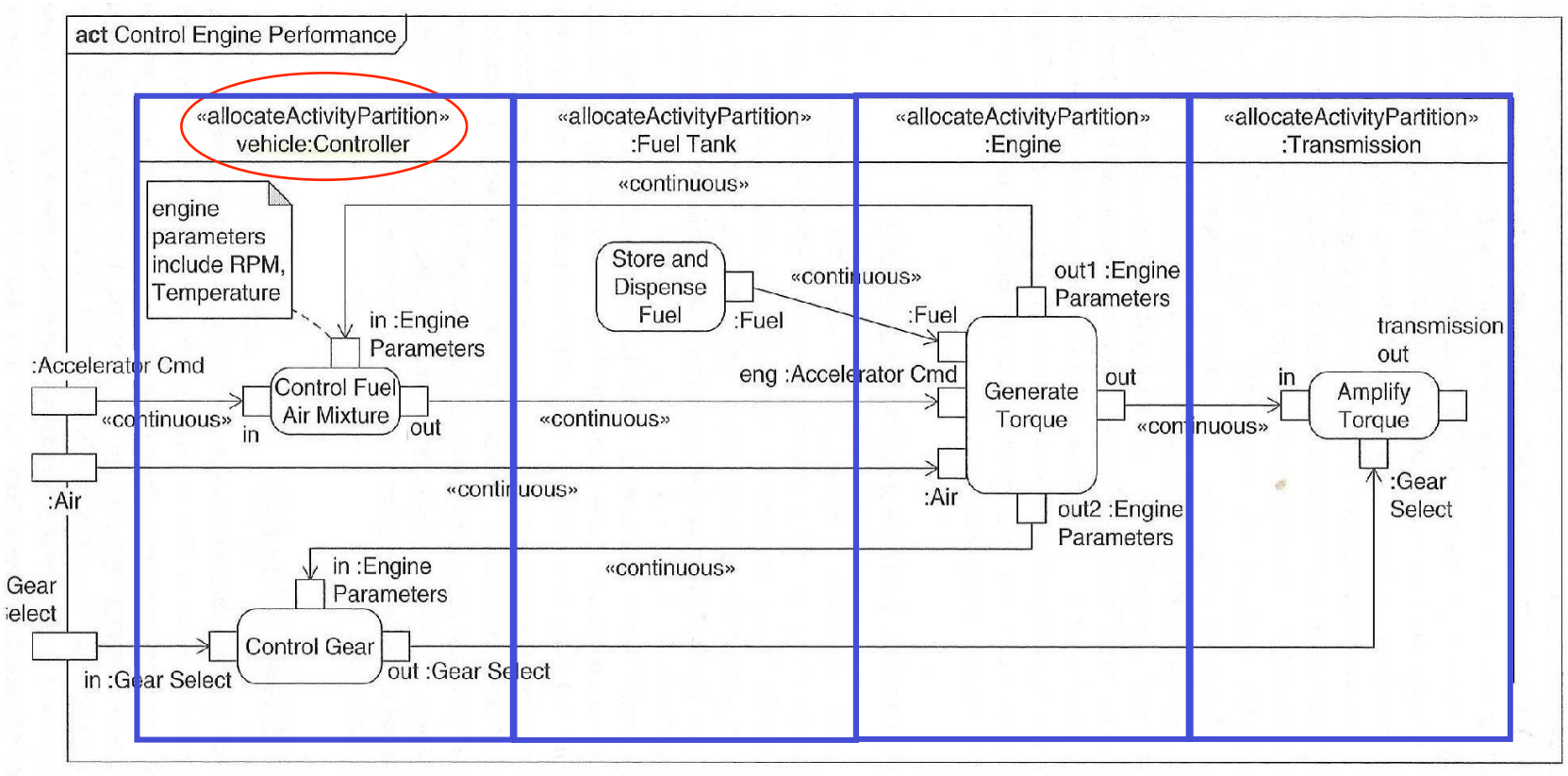
Activity diagram example



Activity diagram notation



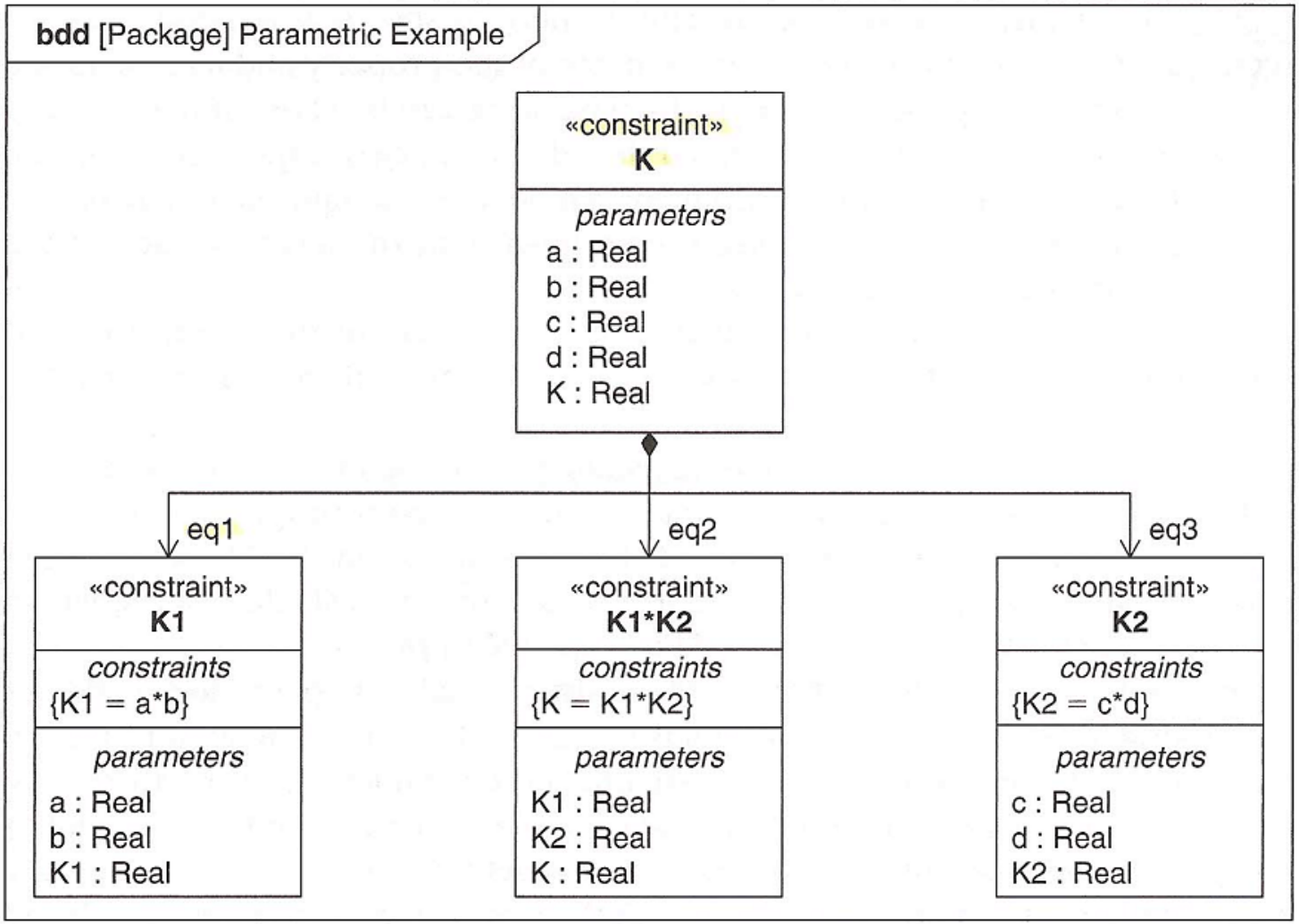
Activity diagram with swimlanes



4. SysML Parametric diagrams

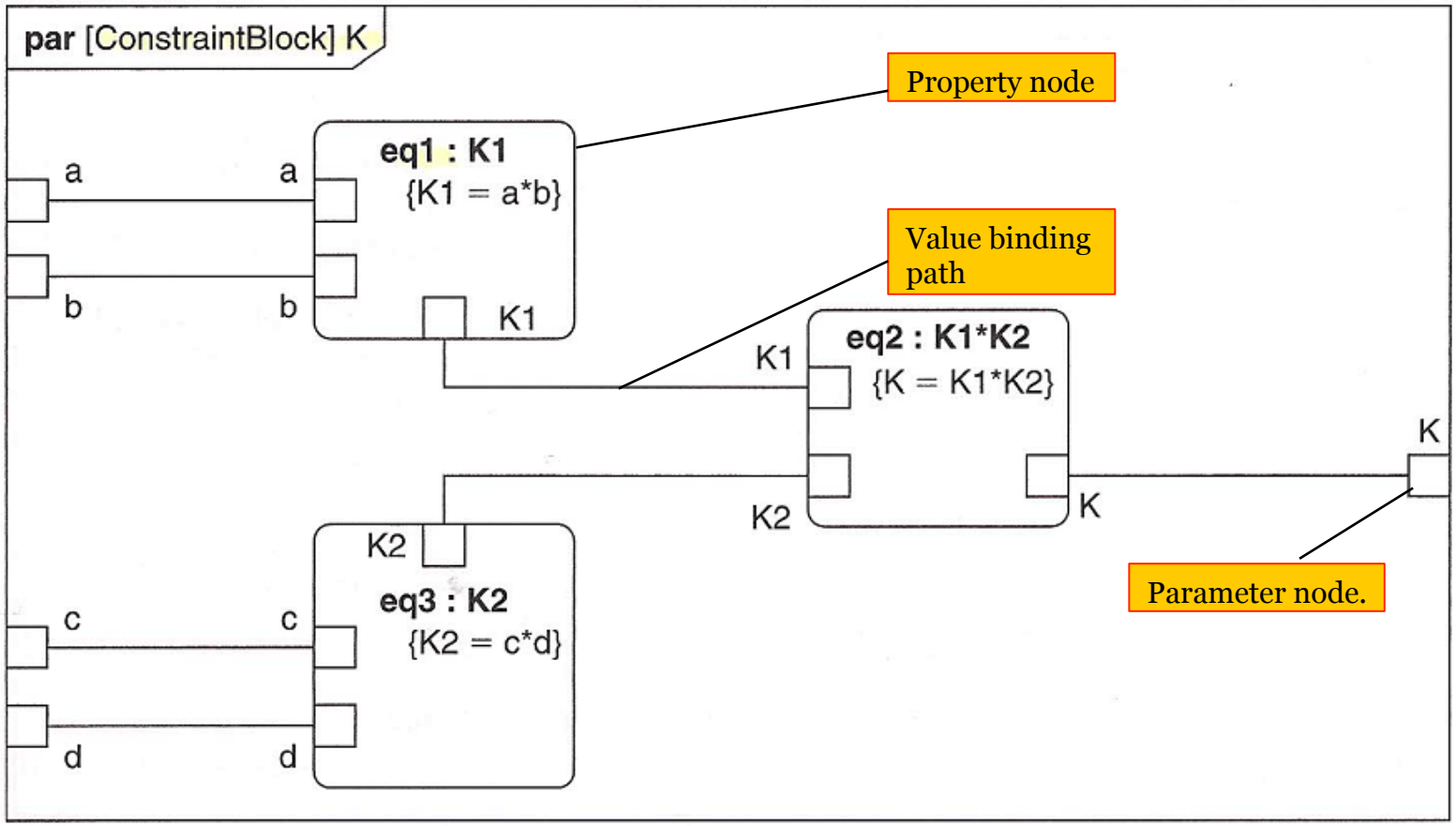
- Used to express constraints (equations) between value properties
  - Provides support for engineering analysis (e.g., performance, reliability)
- Constraint block captures equations shown on a bdd
  - Expression language can be **formal** (e.g., MathML, OCL) or **informal.**
  - Computational engine is defined by applicable analysis tool and not by SysML.
- Parametric diagram represents the usage of the constraints in an analysis context
  - Binding of constraint usage to value properties of blocks (e.g., vehicle mass bound to F= m × a)
- Parametric diagrams enable the integration of engineering analysis with design models.

<< ConstraintBlock >>

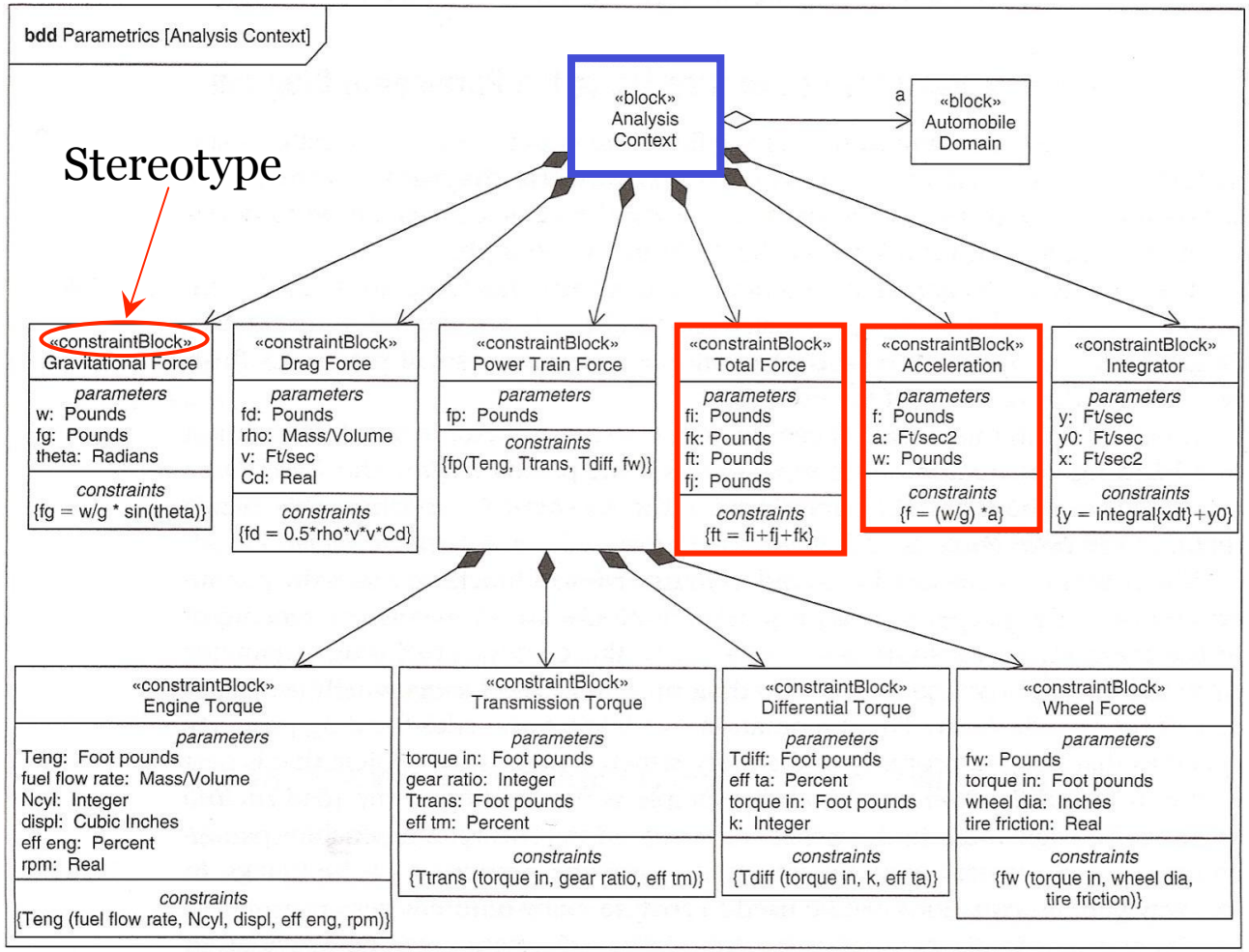


Parametric Diagram

- Example for a constraint block – binding a,b,c,d with K

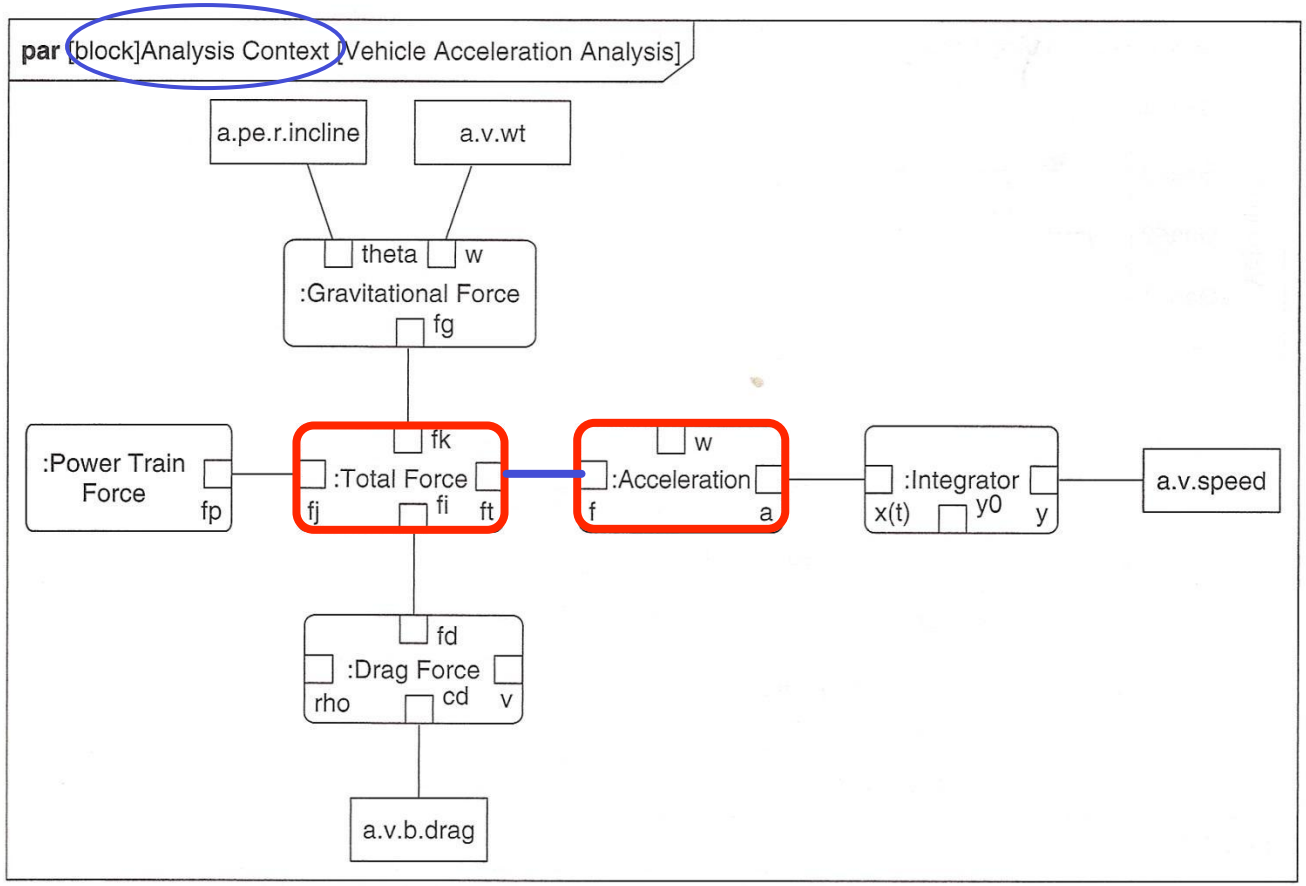


Parametric Constraint Blocks



Parametric Diagram

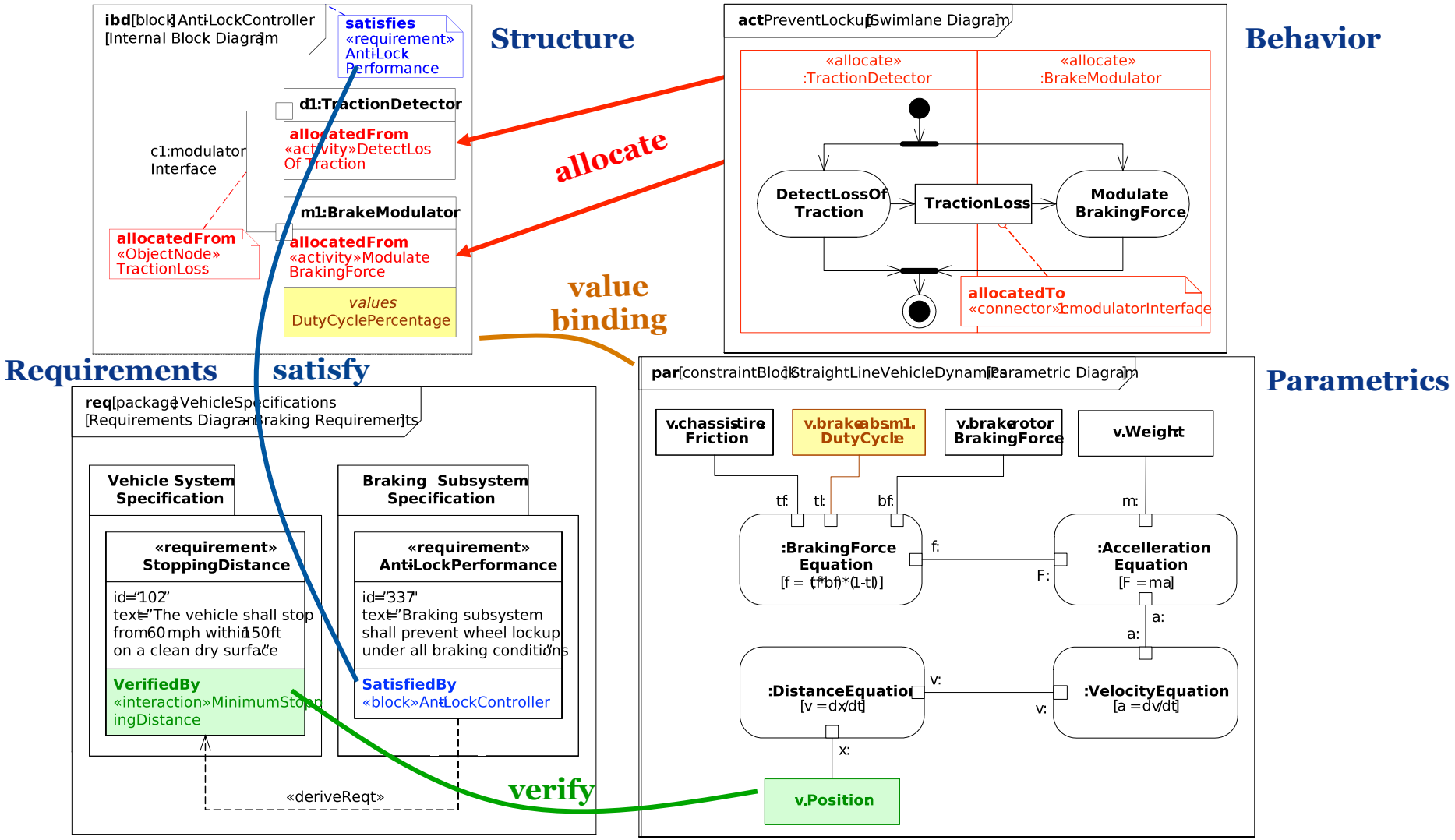
Example for a block



Allocations

Kind of allocations	Relationships	From	To
Requirement allocation	Satisfy	Requirement	Model element
	DeriveReqt	Requirement	Requirement
	Refine	Model element	Requirement
Functional allocation	Allocate	Activity Action	Block Part
Structural allocation (e.g., logical to physical)	Allocate	Block Port Item flow Connector	Block Port Item flow Parts and connectors
Flow allocation	Allocate	Object flow Object flow Object flow	Connector Item flow Item property
Property decomposition/ allocation	Binding connector	Value property	Parameter

Cross-connecting model elements



SysML tools

- ARTiSAN Software Tools (multiuser and free single user version)
- IBM Rational® Rhapsody® Developer is a UML/SysML-based, model-driven development environment for real-time and embedded systems and software
- EmbeddedPlus Engineering (third party tool for IBM Rational)
- Magic Draw – SysML Plugin
- Papyrus for SysML (open source eclipse modeling tool)
- Software Stencils - Microsoft Visio SysML and UML templates
- Sparx Systems Enterprise Architects (supports SysML)

Perspectives for SysML

- Enable a common modeling language and model across engineering disciplines
- Enable traceability between disciplines
- Enable different kinds of system analysis
- Enable integration of discrete and continuous based modeling tools
- Critical enabler for Model-Based System Engineering with tool support

Mandatory exercise – pill dispenser

› BDD and IBD of pill dispenser

Summary

- SysML a common modeling language for different disciplines e.g., hardware, software and mechanics.
- New and important concepts for cross-disciplinary analysis of system properties (e.g. parametric).
- Blocks and ports as general modeling elements.
- Important enhancement to activity diagrams.
- Lot of support for traceability between models and model elements.
- Must be supported by an appropriate systems engineering process.

References

- OMGs SysML homepage: www.omgsysml.org
- *"An Overview of the Systems Modeling Language for Products and Systems Development"* by L. Balmelli, Journal of Object Technology, Vol. 6, No. 6, July-August 2007.
- IBM Rational Harmony: http://www-01.ibm.com/software/rational/services/harmony/
- *Books:*
  - *"A Practical Guide to SysML – The System Modeling Language"*, Sanford Friedenthal, Allan Moore, Rick Steiner, Elsevier, 2009.
  - *"Systems Engineering with SysML/UML – Modeling, Analysis, Design"*, Tim Weilkiens, Elsevier, 2007.