Introducing Object Oriented Systems Engineering Methods to University Systems Engineering Curricula

Chris Ryder

Johns Hopkins University Applied Physics Laboratory

NATIONAL DEFENSE INDUSTRIAL ASSOCIATION
NDIA
STRENGTH THROUGH INDUSTRY & TECHNOLOGY

JOHNS HOPKINS ENGINEERING | The Whiting School of Engineering

APL
The Johns Hopkins University
APPLIED PHYSICS LABORATORY

References

- *Systems Engineering: Principles and Practices*
  - Alexander Kossiakoff and William Sweet
- OMG SysML Tutorial (Presented by Abe Meilich on 10/23)
- Sandy Friedenthal – "Model Driven Architecture for Systems Engineering"
- http://www.omgsysml.com
- OMG SysML Specification

Observation

- There is no standardized approach to SE architecting, modeling and design used in the JHU WSE SE Curriculum
  - Most instructors teach the methods they are familiar with
- The single common element is the SE Method
  - And its relationship to SE Life Cycle and Materialization
- Most students/ classes use Power Point as the modeling tool
  - Difficult to portray engineering diagrams
  - Does not “contain” any data details
- Tools used by instructors include:
  - VITECH Core
  - Sparx Enterprise Architect
  - MS Visio

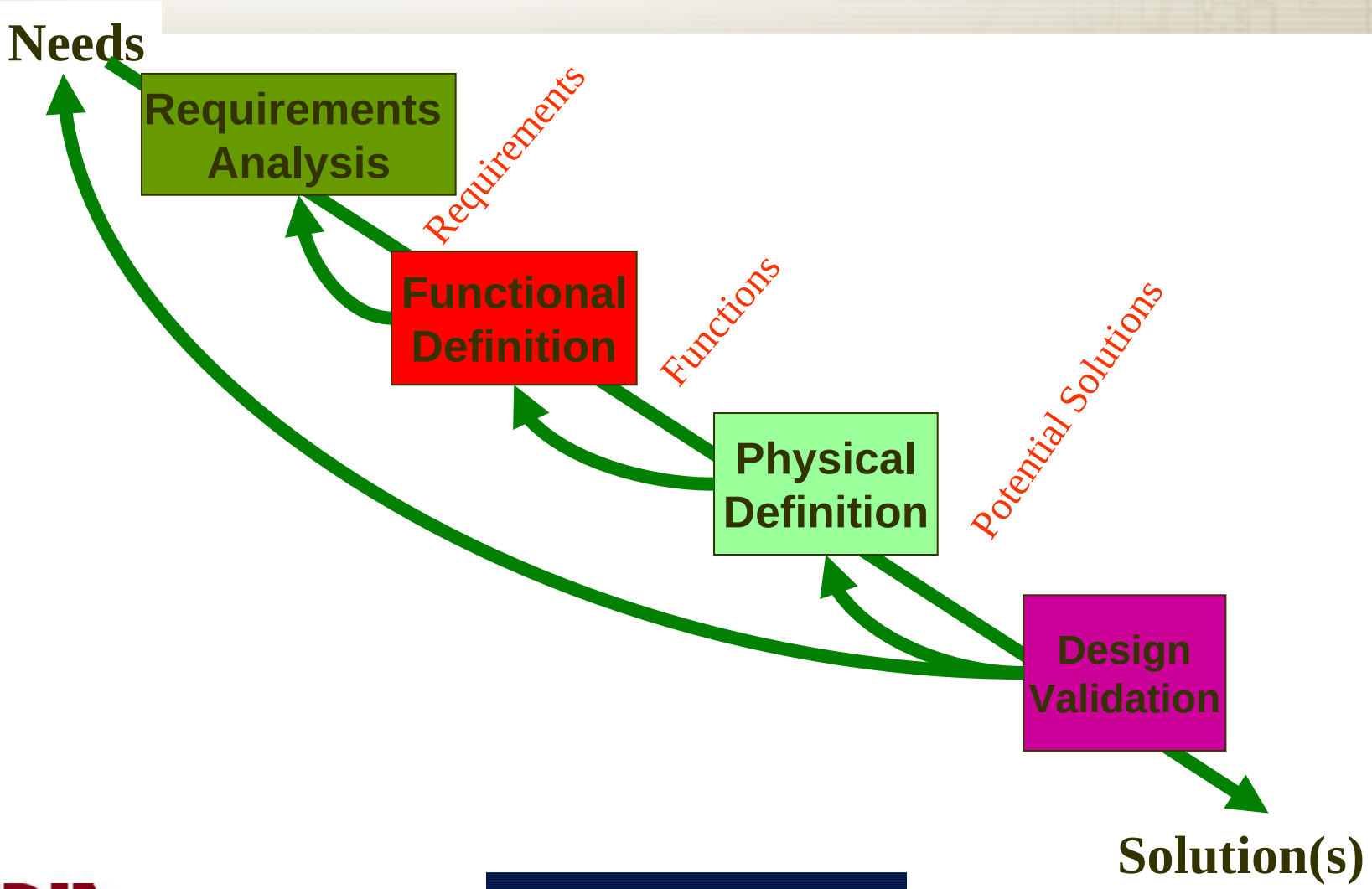
Proposition

- Introduce Object Oriented Systems Engineering Methods as the basis for architecting, modeling and design for design related activities in SE courses
- At a minimum, introduce model-driven SE using a standardized modeling language
  - Independent of method
  - Independent of any specific tool
- OMG SysML meets this criterion
  - It is standardized (released by OMG on 6 July 06)
  - Implemented by several tool vendors
    - Most of whom will provide licenses at little (i.e. < $100) or no cost

The Systems Engineering Method

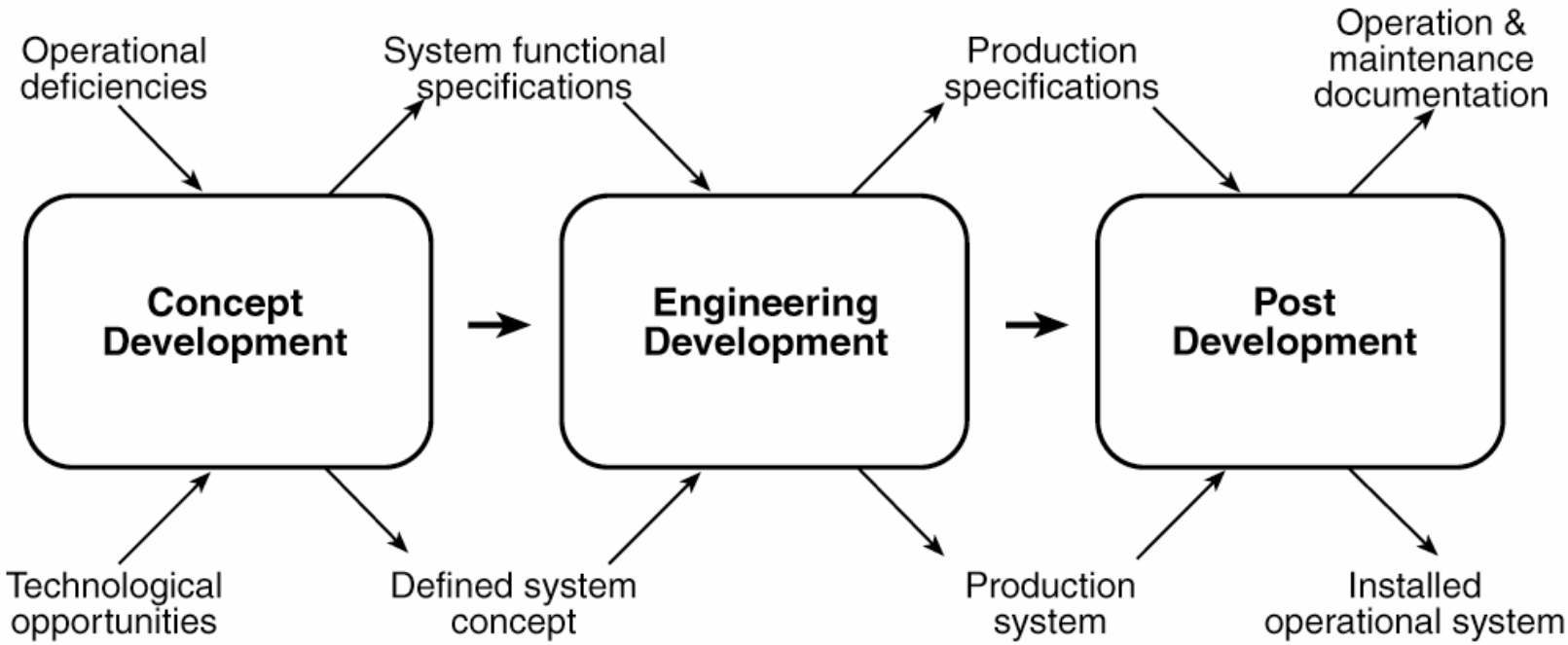
- Every phase of the systems life cycle consists of some form of:
  - Requirements Analysis
  - Functional Definition
  - Physical Definition
  - Design Validation
- This is the basis of the JHU WSE Systems Engineering curriculum
- The SE Method is applicable to both traditional Structured Analysis or with OOSEM

Systems Engineering Method



Principal Stages in System Life Cycle (Kossiakoff & Sweet)

Products



System Models (Baselines)

System Materialization

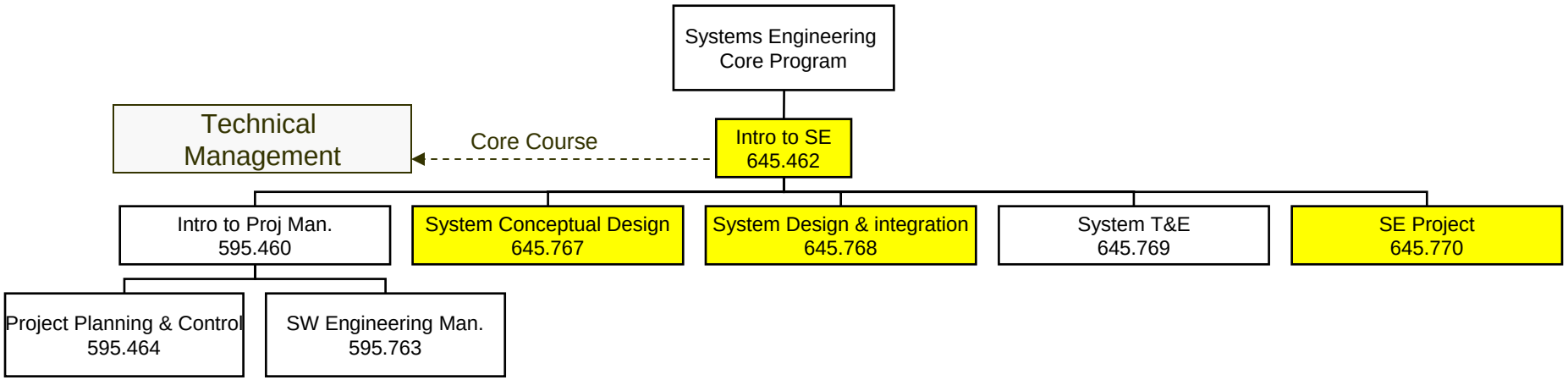
Phase/Level	Needs Analysis	Concept Exploration	Concept Definition	Advanced Development	Engineering Design	Integration and Evaluation
System	Define Operational Objectives	Explore Concepts	Define Selected Concepts	Validate Concept		Test and Evaluate (System and Operational)
Subsystem	Visualize	Define Functions	Define Configuration	Validate Selected Subsystems		Integrate, Test
Component		Visualize	Select, Define Functions	Validate, Specify construction	Design, Test	Integrate,
Sub-component			Visualize	Define Functions	Design	
Part				Visualize	Select or adapt	

Ref: SYSTEMS ENGINEERING PRINCIPLES & PRACTICE
A Guide to the Engineering of Complex Systems

JHU WSE Systems Engineering Program

- ~400 student enrolled
- Curricula offered at four primary campuses
  - APL, JHU Montgomery County Campus, WSE Dorsey Center, Southern Maryland Higher Education Center
- Curricula also offered on site at industry locations
  - MITRE (Bedford, MA; Vienna, VA)
  - NAVSEA (Crystal City, VA)
  - BAE Systems (Nashua, NH)
- Courses conducted by instructor teams
  - One from APL and one from industry

JHU EPP
SE Core Program



- Denotes courses with modeling and design projects

Why Object Oriented SE?

- Applies the SE Method
  - Requirements are captured using SysML requirements diagram and Requirements Traceability Matrix (RTM)
  - Functional analysis and decomposition performed using SysML behavioral diagrams
  - Physical elements, behaviors and relationships are modeled using SysML structural diagrams
    - Functionality assigned to physical objects
- Trace model elements to requirements

OOSE Requirements Capture

- Start at the Beginning
  - Needs Analysis and requirements definition
- Formulating the “Requirements Model”
  - Define/scope the problem
  - Analyze requirements
    - Necessary, concise, attainable, complete, consistent, unambiguous, verifiable
    - Create requirements traceability
  - Documenting the requirements
    - Constructing the SysML Requirements Diagram
    - Building the Requirements Traceability Matrix

OOSE Functional Analysis

- **Structured SE and Object Oriented SE Methods are "Homeomorphic"** (Joe Carl, PhD, Retired Guy)
  - **"Possessing intrinsic topological equivalence"**
  - **SA representation can be directly mapped to OO form**
- **In other words:**
- **OO methods involves the same "Top Down" hierarchical approach**
  - **Top Down/ Breadth First**
  - **Event Driven**
    - **Objects have well-defined functionality that execute tasks as a sequence of events**
    - **Interactions between objects are defined at each level in the system**
    - **Refined as lower-level objects become instantiated**
    - **Systems, subsystems, components exist in a state**

Object Oriented

- Every system is composed of “Objects”
- All Objects contain Attributes, Operations, Parameters and Constraints
  - Operations → FUNCTIONS
- Functional analysis still applies to OOSE
  - Operations are assigned to an object, however abstract, early in the process
  - Unlike with OOSWE, “Functional Decomposition” is not a dirty word
- Measures are contained within the objects
  - Measures that can quantify objectives

So What is the Difference?

- Focus on the “Logical” as opposed to the “Functional”
  - Logical elements posses both Function and State
- Analyze the system from the viewpoint of the “Things” however abstract
  - Consistent with Systems Materialization
- OOSE is a model-driven methodology by definition

Systems Engineering Model

- A Systems Engineering model captures the essential elements of the systems engineering life-cycle
- "Dynamic and recursive process" (Bootch, Rumbaugh, Jacobson)
  - Iteratively captures enterprise capabilities and systems requirements
  - Promotes incorporation of technology evolution
- Forms basis for sound, long-term SE and analysis
  - Compliant with DoDAF and JCIDS

Model-Driven SE Approach

- Establish system model bases on:
  - Requirements model
  - Functional model
  - Logical/ Physical model
- Show relationships between the models
  - Link requirements to functions
  - Link functions to system/ elements
- Understand the capability being developed

"If you don't model it, you won't understand it."
Ivar Jacobson

OMG SysML

- OMG SysML™ is a standardized family of diagrams that addresses requirements, functional and logical/physical elements
  - OMG SysML is suitable for both OO and structured methods, but it was formulated from UML with OO methods in mind
    - SysML standard ~ 230 pages
  - As opposed to non-standard SA diagrams
    - IDEF-0 (180 Pages)
    - IDEF-3 (235 Pages)
    - Data Flow/ Control Flow (No standard)
    - Functional Flow Diagrams (No standard)
    - Enhanced Functional Flow Block Diagrams (No Standard)
- Tool Vendors are implementing it in their applications

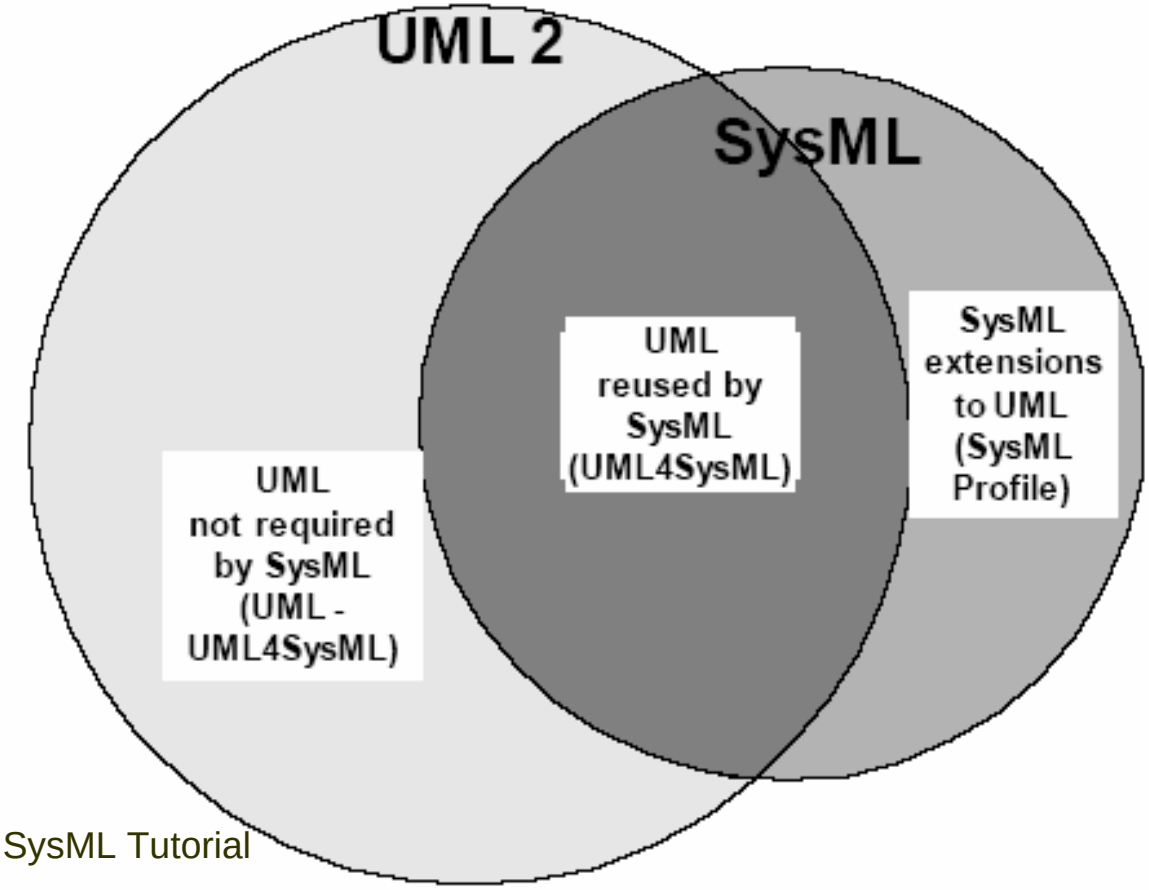
What Is SysML?

- A graphical modeling language in response to the UML for Systems Engineering RFP developed by the OMG, INCOSE, and AP233
  - A UML Profile that represents a subset of UML 2 with 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 and the evolving AP233 standard (in-process)

SysML is Critical Enabler for Model Driven SE

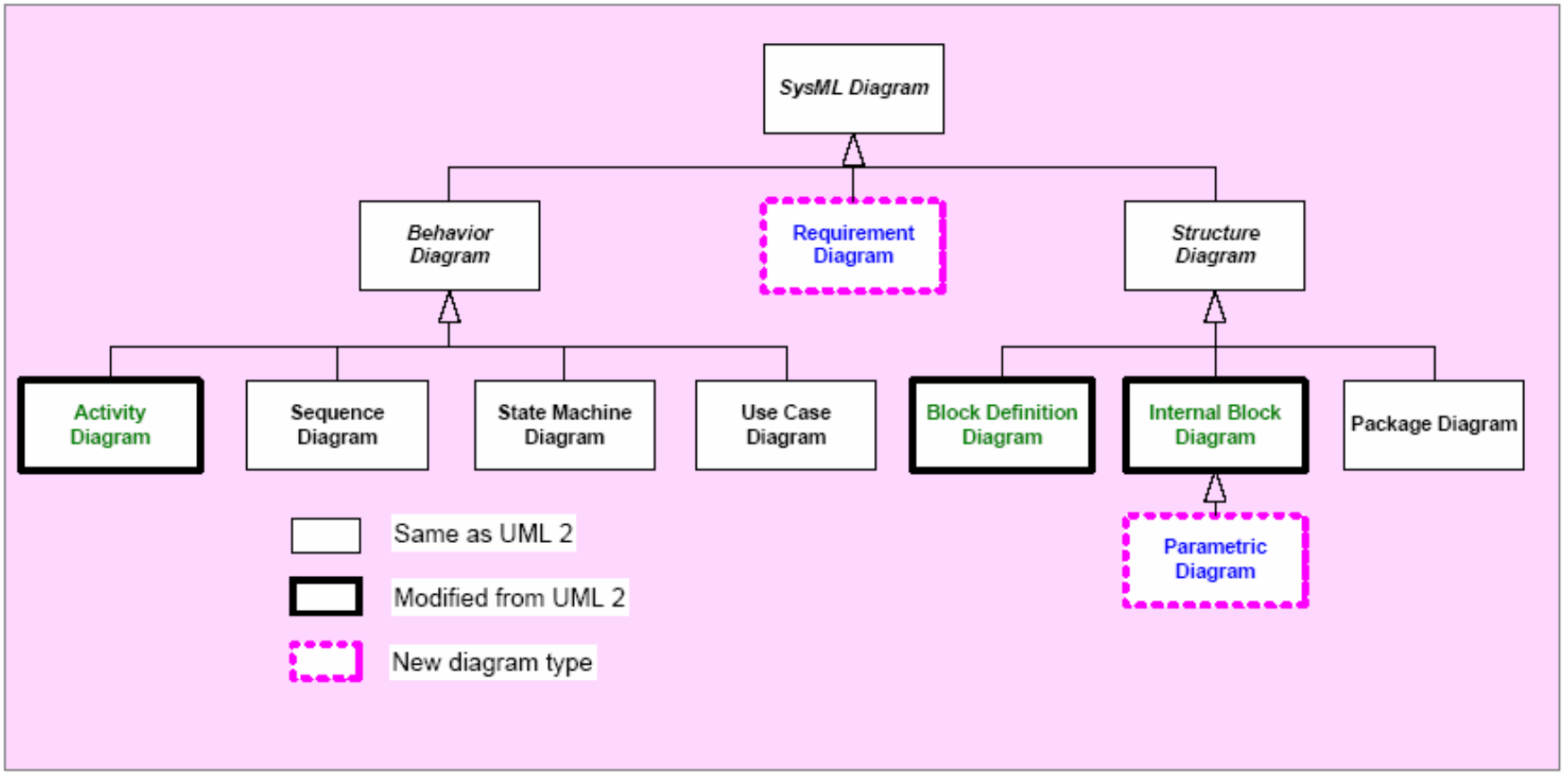
Source: OMG SysML Tutorial

Relationship Between SysML and UML



Source: OMG SysML Tutorial

SysML Taxonomy

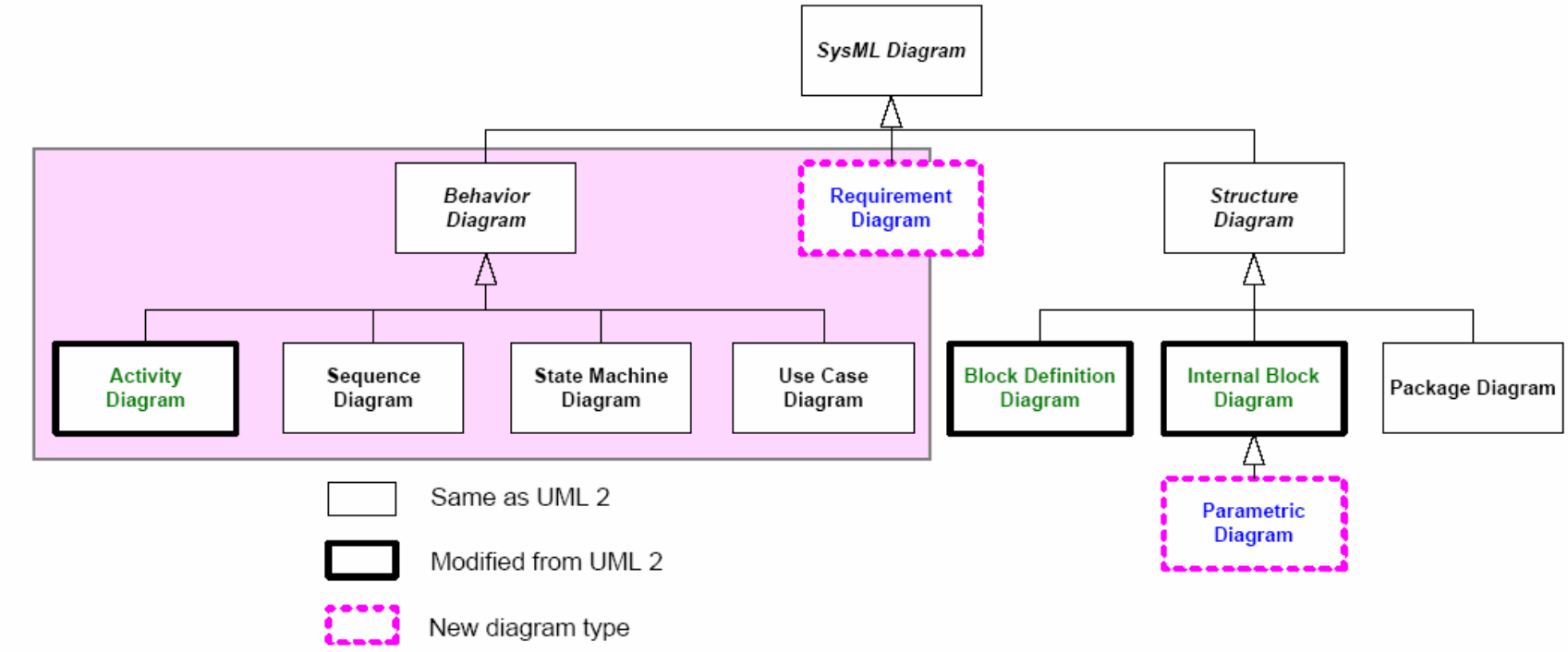


Source: OMG SysML Tutorial

Behavioral Elements in the Functional Model

- Represented by Use Cases and SysML behavioral diagrams
- Executed by “Actors” outside the system boundary
  - Actor is a form of Block and its own attributes and operations
- Actor represents a role, not an person or group
- Block at one level can be actor at an other
  - Actors are often external systems or internal system controls
- Actor executes the Use Case on the materiel object, i.e. the system, subsystem, component or part

SysML Behavioral Diagrams



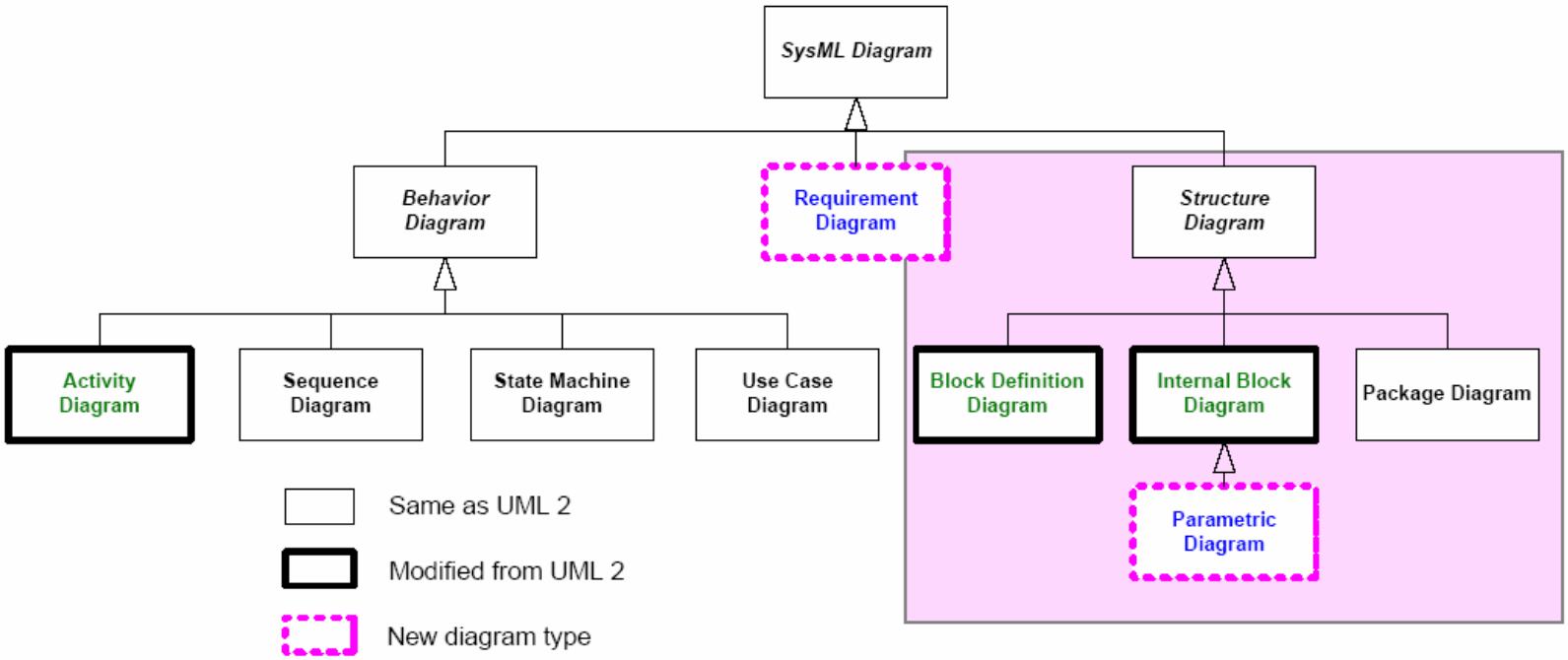
Source: OMG SysML Tutorial

The Logical Model

- Physical Definition
  - Beginning with abstract “things”
  - Evolve to real systems
- Assign functionality to the element
- Depict the relationships between elements

**The Logical Model is the heart of an architecture –
Elements that exhibit behavior and their defined relationships with other elements within the domain**

SysML Structural Diagrams



Source: OMG SysML Tutorial

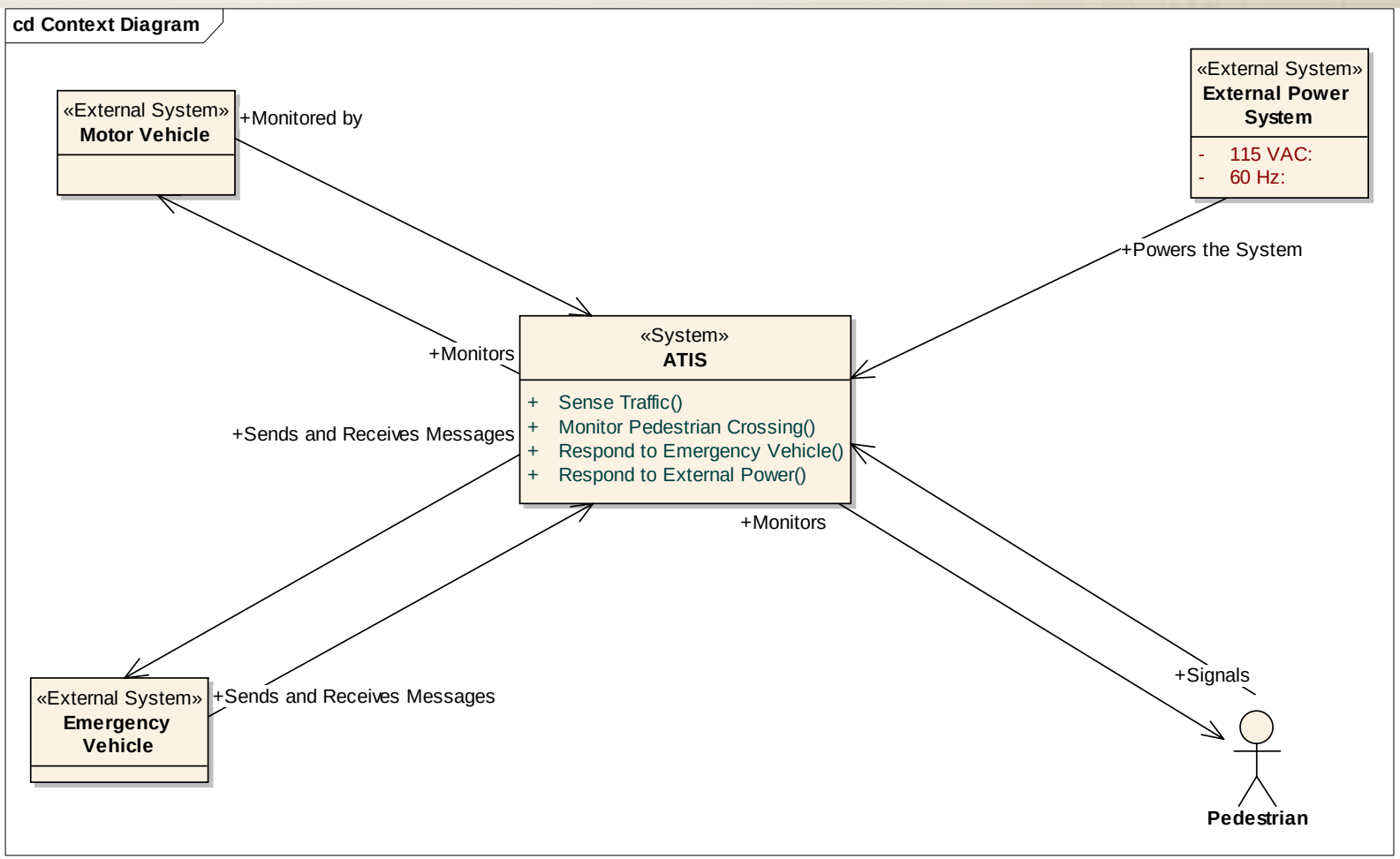
Structural/ Physical Elements

- Depicts basic logical structure of the system
  - Packages
    - Organize the elements as sub-entities
  - Blocks
    - Basic structural element
      - Same specification as the UML Class
      - Consists of attributes, operations, associations, constraints
      - Also represents human and organizational elements –- the Actor
  - Ports
    - Specifies interaction points or parts
    - Specifies flow or standardized interface
  - Parametrics
    - Specifies constraints with value types

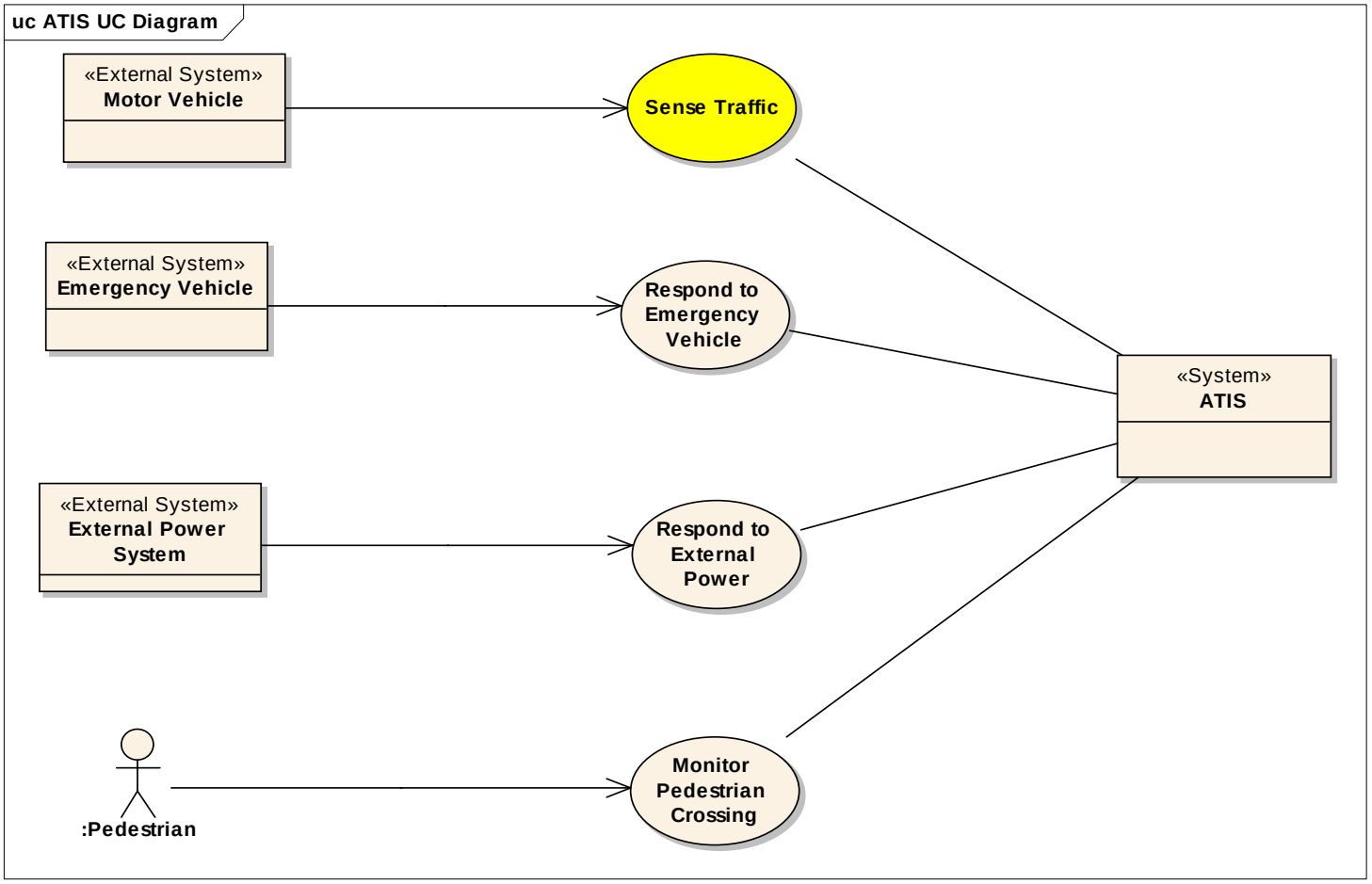
ATIS Case

- Automated Traffic Intersection System
- Students presented a set of “less-than-good” requirements
  - Describe what improvements need to be done
- Describe the “top level” functions
  - Initiate functional analysis
- Describe the logical elements with assigned functionality
- Depict a hierarchy of components with functions
- Consider interfaces for one subsystem

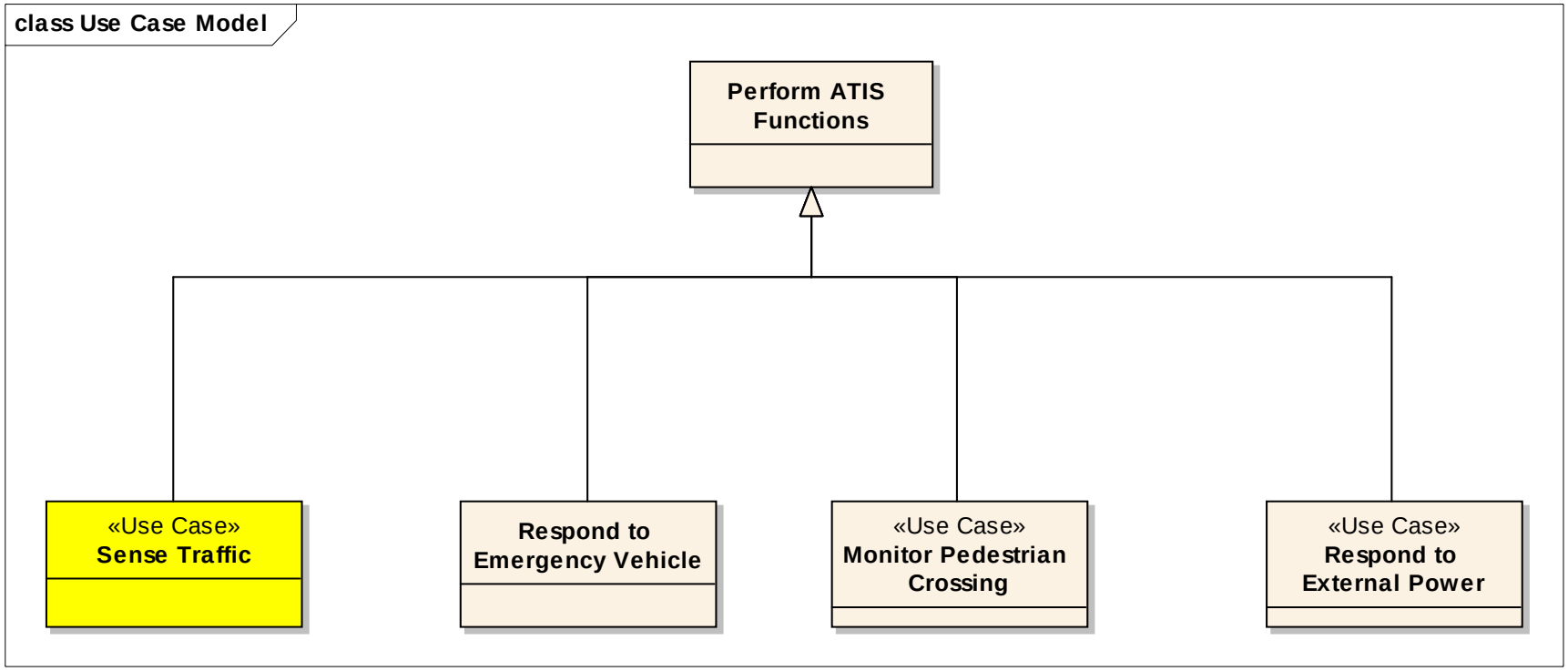
ATIS Context Diagram



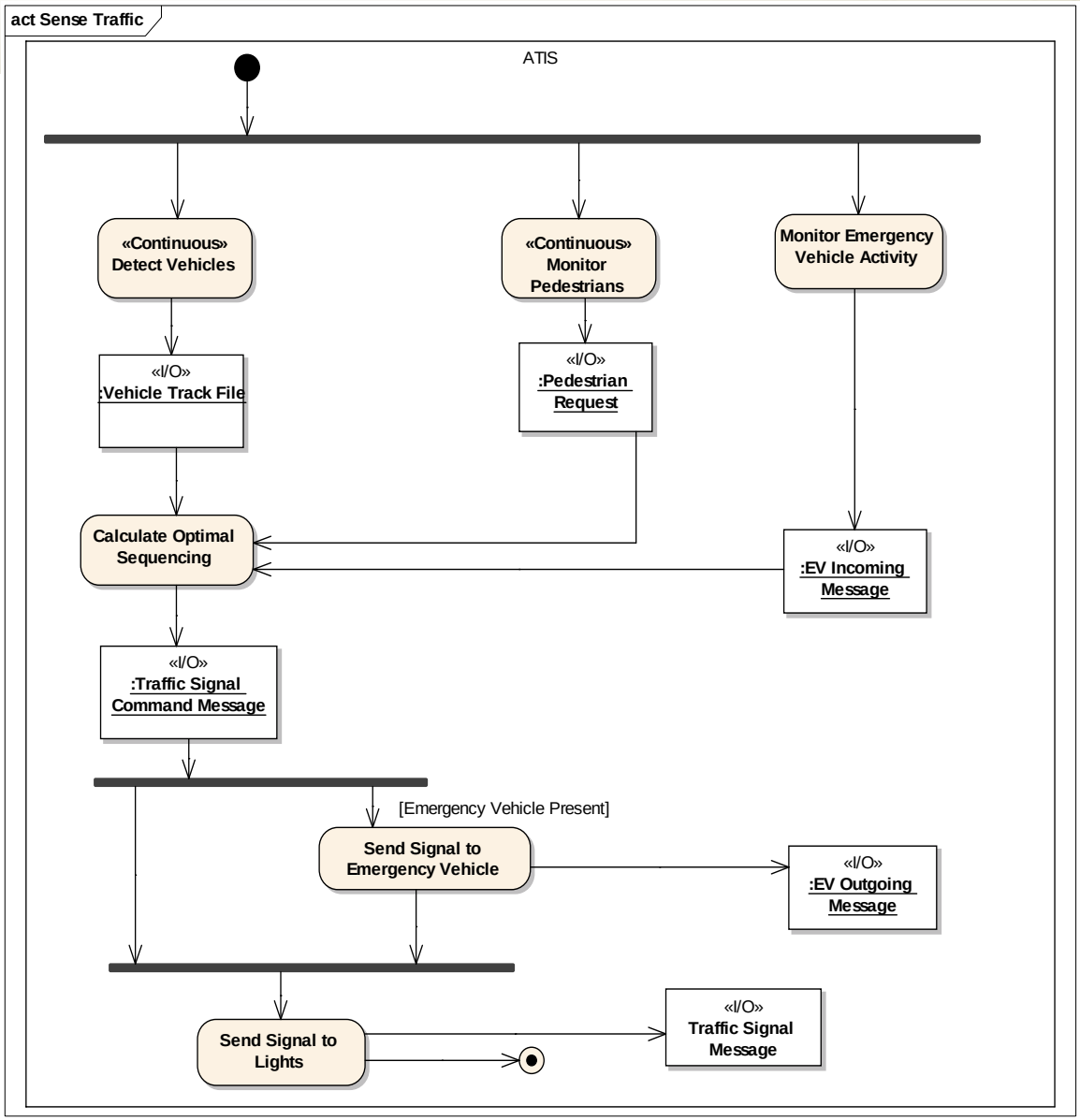
ATIS Use Case Diagram



Functionality Depicted in Hierarchical Form

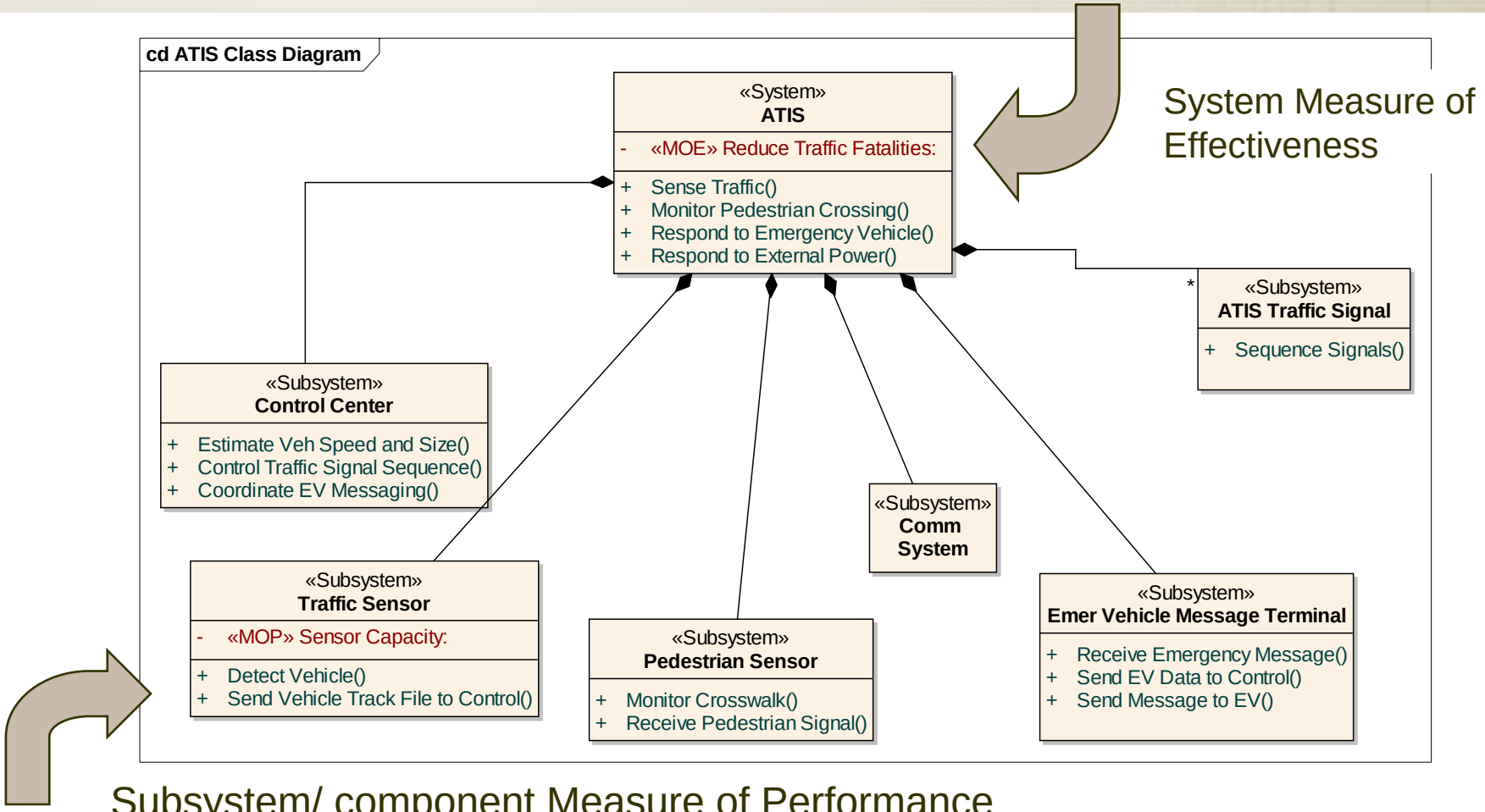


ATIS SysML Activity Diagram

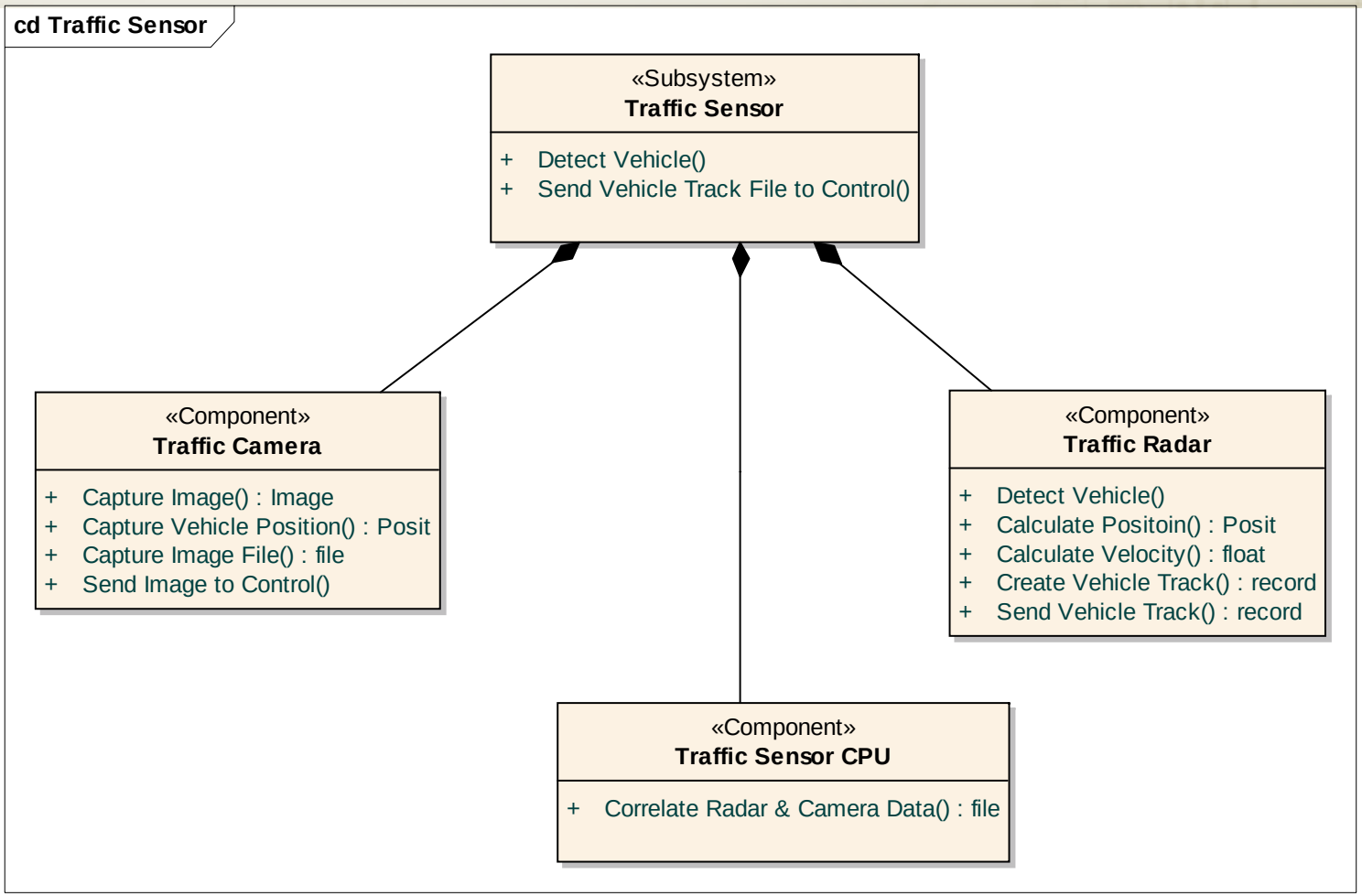


ATIS Block Definition Diagram

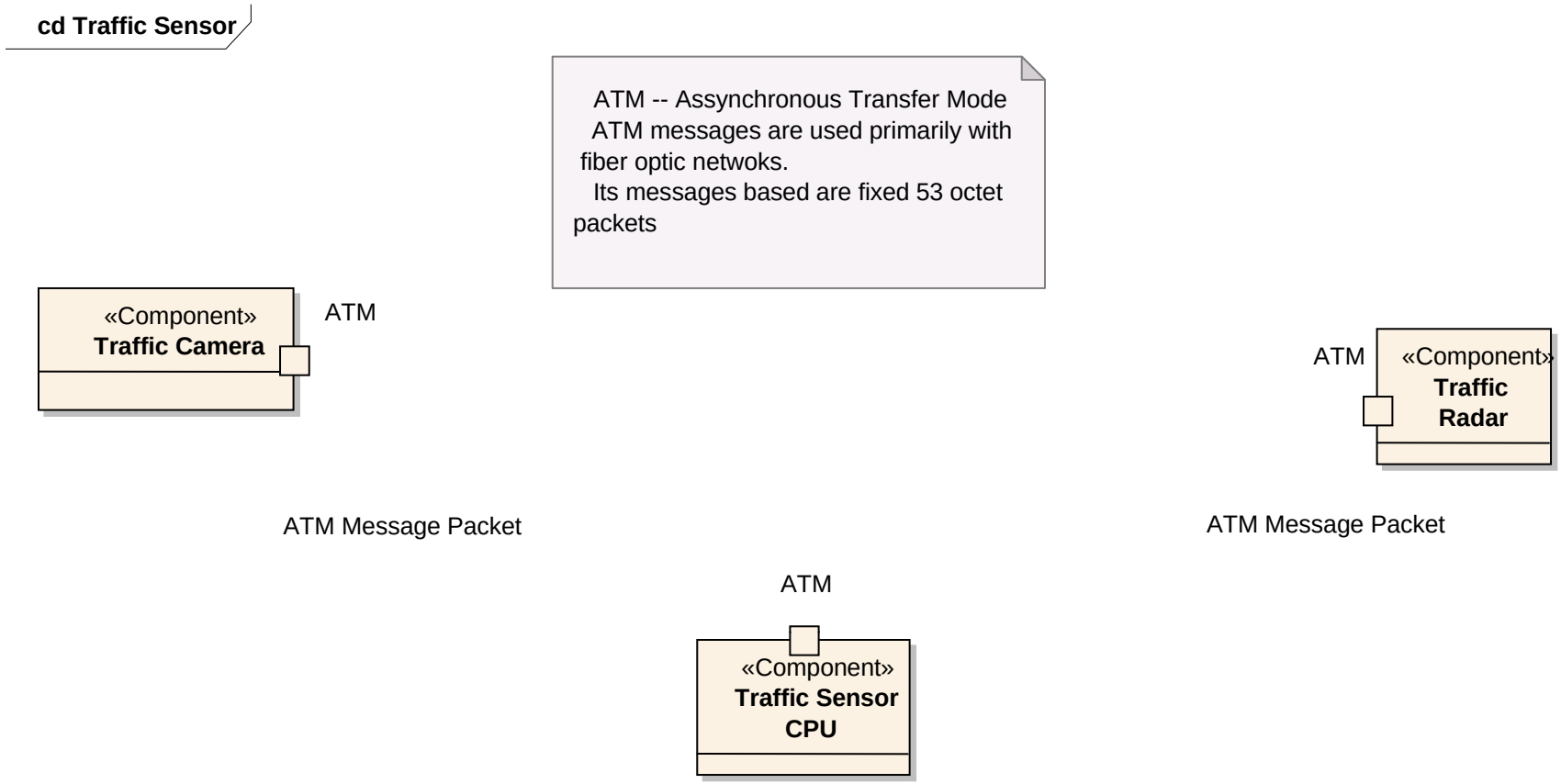
At the system level, attributes must be measurable!



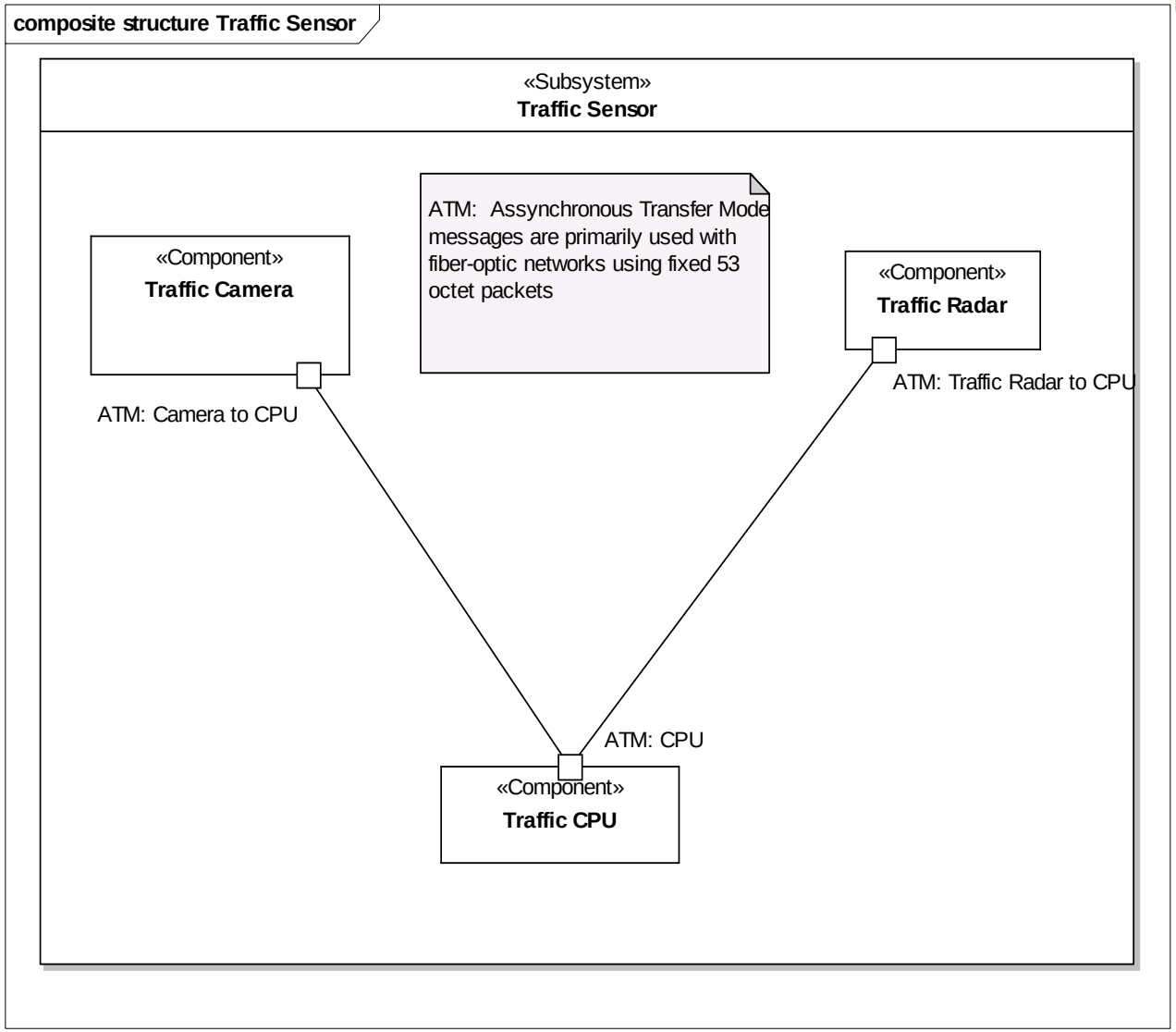
ATIS Traffic Sensor Subsystem



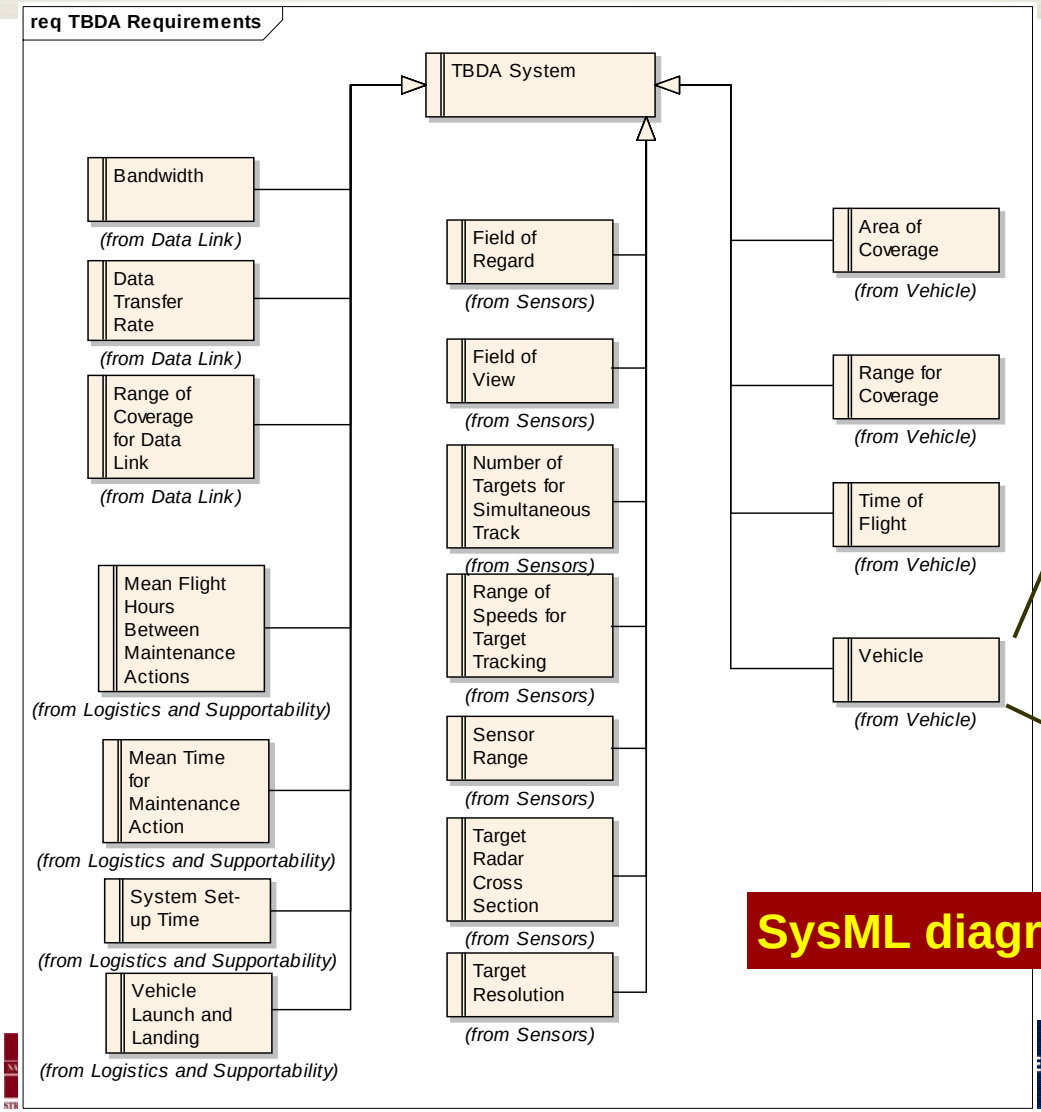
Example of Internal Block Diagram

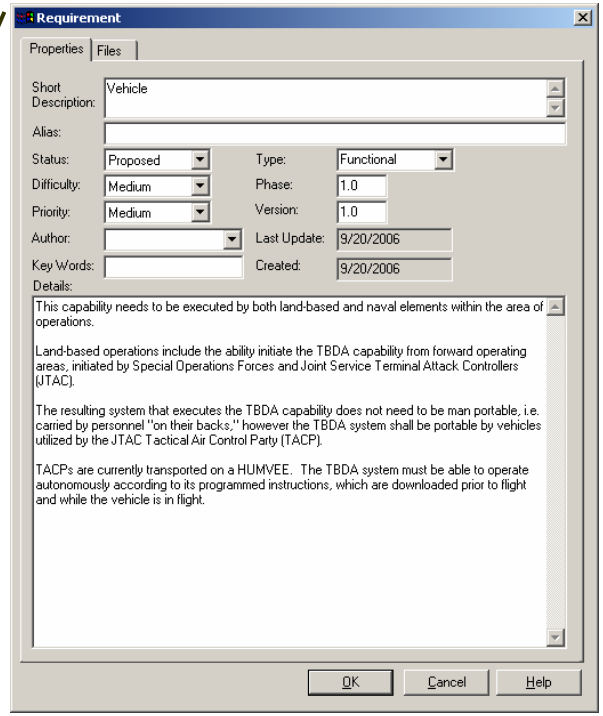


Example of Internal Block Diagram



TBDA Requirements Diagram





SysML diagrams supported by robust database

Engineering

APL

Summary of OOSE

- There is nothing special about using Object Oriented SE methods
- It involves the same basic analysis
- OOSE is a model-driven style
  - Models are fundamental to architecture development
- Human beings think in terms of “things”

Conclusion

- This proposal considers only an introduction to basic OOSEM practices
  - Details of OOSEM is far beyond the scope of design courses
  - OOSEM using SysML could be an entire semester course
  - The INCOSE tutorial is intended to introduce detailed practices for real-world project usage
- By introducing OOSE principles at the University, students can apply the SE Method as it relates to Systems Materialization across the life-cycle
- Standardized modeling methods must be applied
  - Instructors must keep up with evolving industry practices
  - Observation from INCOSE 06 and NDIA SE Conference indicates SysML will be widely used throughout industry

If anything else, you know what "Homeomorphic" means