

CSSE 490 Model-Based Software Engineering: AADL and SysML



Shawn Bohner

Office: Moench Room F212

Phone: (812) 877-8685

Email: bohner@rose-hulman.edu



ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

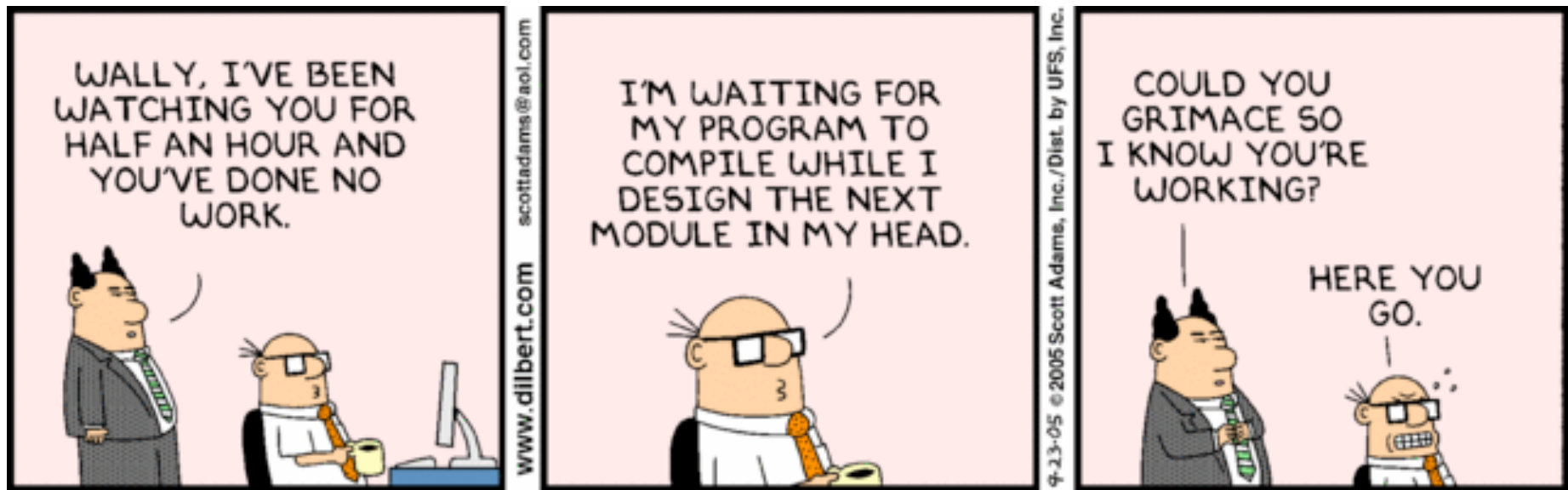
Learning Outcomes: MBE Discipline

Relate Model-Based Engineering as an engineering discipline.

- Demonstration of Milestone 3
- Introduce AADL
- Compare SysML



So, how is Milestone 3 coming 😊?



How can Architecture Description Languages (ADL) help in the automated generation of software?

- Think for 15 seconds...
- Let's talk...



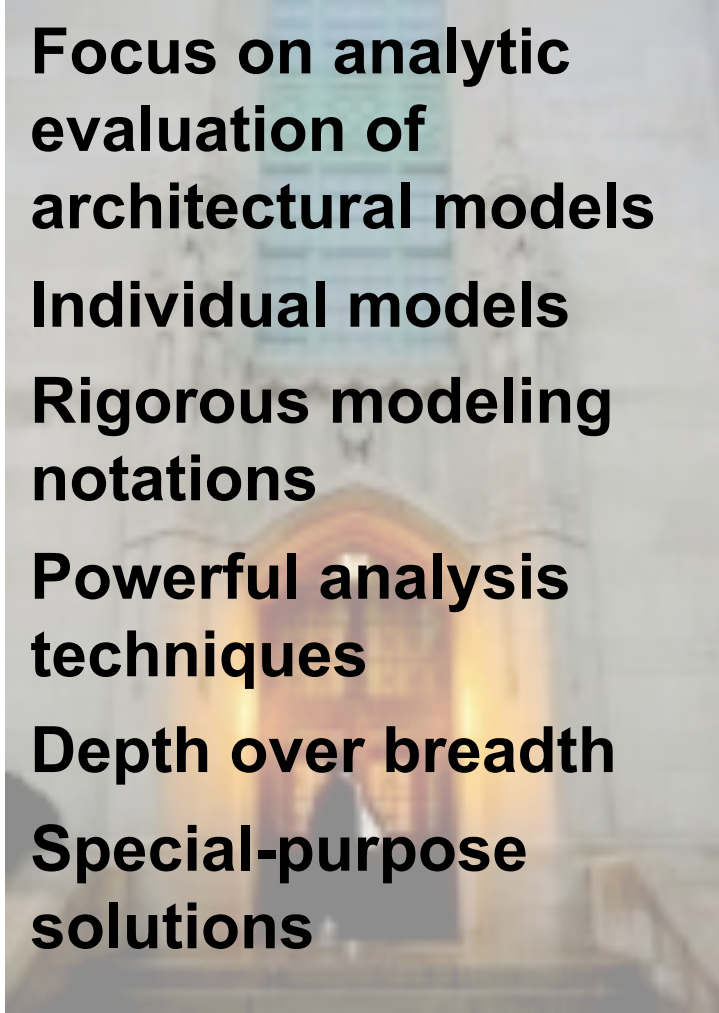


Approaches to Architecture

Industrial Approach

- 
- Focus on wide range of development issues
 - Families of models
 - Practicality over rigor
 - Architecture as the “big picture” in development
 - Breadth over depth
 - General-purpose solutions

Academic Approach

- 
- Focus on analytic evaluation of architectural models
 - Individual models
 - Rigorous modeling notations
 - Powerful analysis techniques
 - Depth over breadth
 - Special-purpose solutions

SAE Architecture Analysis & Design Language (AADL) Standard

- Designed for Model-Based Engineering
 - Notation for specification of runtime architecture of real-time, embedded, fault-tolerant, secure, safety-critical, software-intensive systems
- Fields of application:
 - Avionics, Aerospace, Automotive, Autonomous systems, Medical devices ...
- Industry-driven International Standard

■ www.aadl.info

Rockwell
Collins
Honeywell

esa
axlog
INGENIERIE
DASSAULT
AVIATION

EADS
DEFENCE
& SECURITY
AIRBUS
BOEING

LOCKHEED MARTIN
We never forget who we're working for®

Ford

TOYOTA

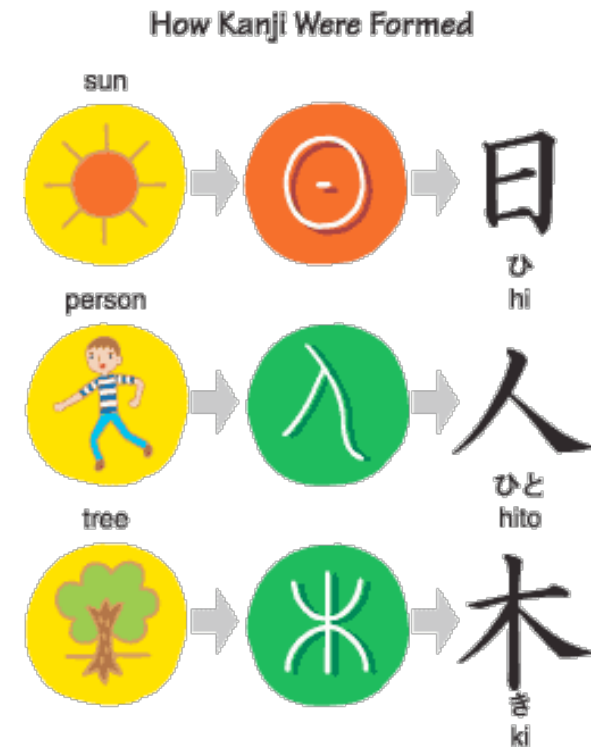


Key Elements of SAE AADL Standard

- **Core AADL language standard (SEI)**
 - Textual & graphical, precise semantics, extensible
- **AADL Meta model & XMI/XML standard (SEI)**
 - Model interchange & tool interoperability
- **UML profile for AADL**
 - Subset of OMG MARTE profile being defined by MARTE
- **Error Model Annex as standardized extension Fault/reliability modeling, hazard analysis**
- **Behavior Annex**
 - Externally observable behavior of components
- **Programming Guidelines, Data Modeling Annexes**

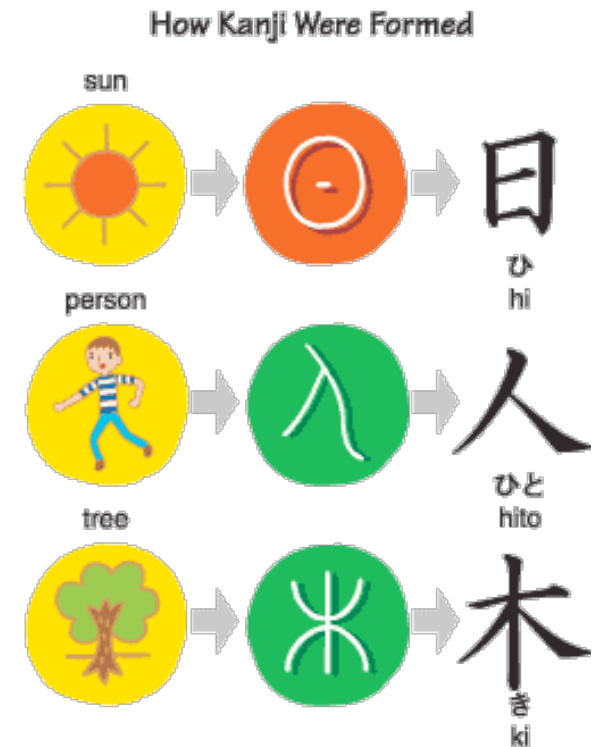
AADL: The Language 1/2

- **Precise execution semantics for components & interactions**
 - Thread, process, data, subprogram, system,
 - Processor, memory, bus, device, abstract component, virtual processor, virtual bus
- **Continuous signal processing & stochastic event processing**
 - Data, event, message communication, unqueued & queued
 - Synchronous call/return, Shared data access
 - End-to-End flow specifications

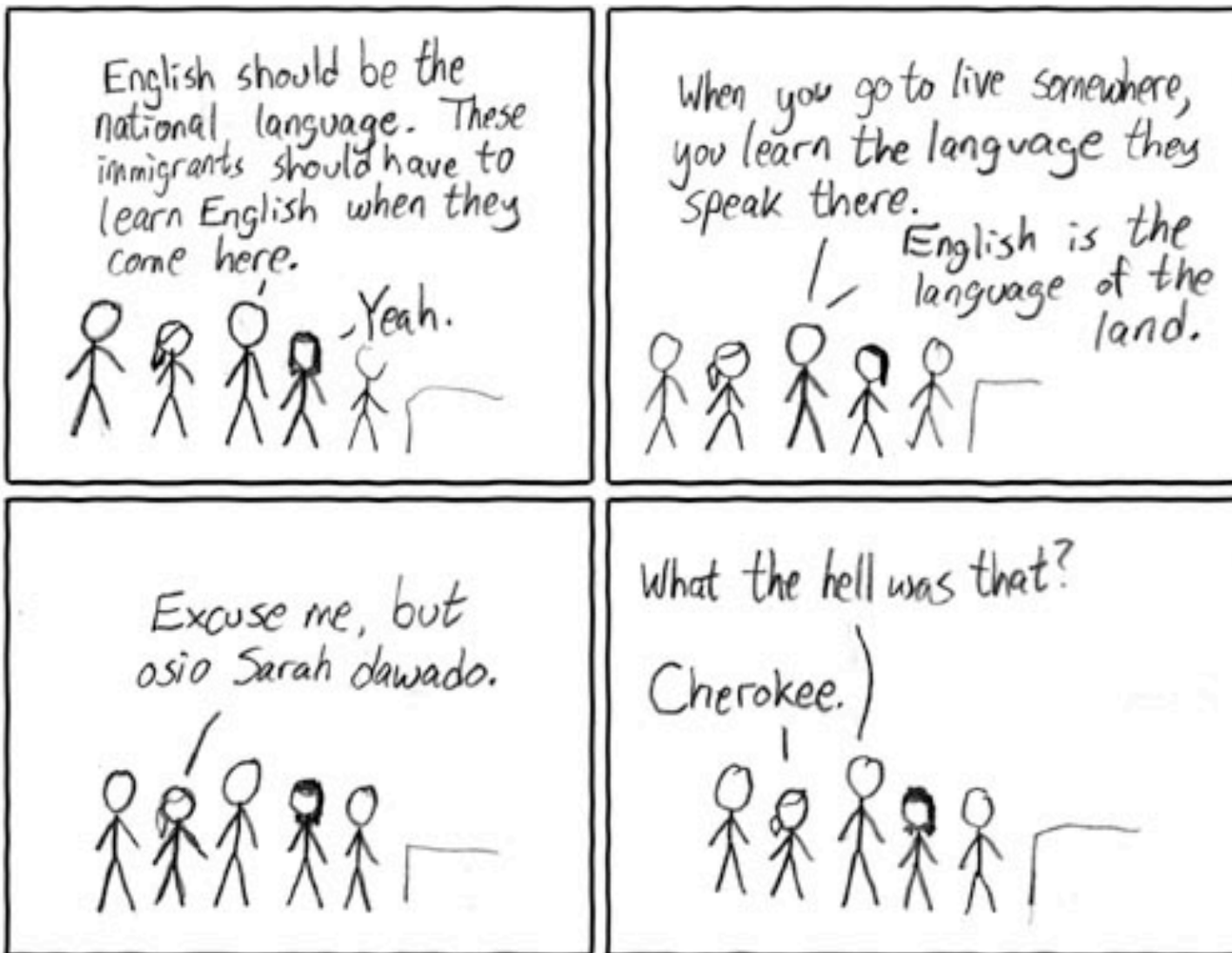


AADL: The Language 2/2

- Operational modes, fault tolerant configurations, levels of service
- Modeling of large-scale and configurable systems
 - Component variants
 - Packaging of component classifiers
 - Layered systems, parameterized templates, component arrays...
- Accommodation of diverse analysis needs
 - User-defined properties, sublanguage extensions



Language Etiquette ...





System Type

System

system GPS

features

```
speed_data: in data port metric_speed
  {SEI::BaseType => UInt16;};
geo_db: requires data access real_time_geoDB;
s_control_data: out data port state_control;
```

flows

```
speed_control: flow path
  speed_data -> s_control_data;
```

```
properties SEI::redundancy => Dual;
end GPS;
```

{type}
extends
features
flows
properties



System Implementation

```
system implementation GPS.secure
subcomponents
  decoder: system PGP_decoder.basic;
  encoder: system PGP_encoder.basic;
  receiver: system GPS_receiver.basic;

connections
  c1: data port speed_data -> decoder.in;
  c2: data port decoder.out -> receiver.in;
  c3: data port receiver.out -> encoder.in;
  c4: data port encoder.out -> s_control_data;

flows
  speed_control: flow path speed_data -> c1 -> decoder.fs1
                 -> c2 -> receiver.fs1 -> c3 -> decoder.fs1
                 -> c4 -> s_control_data;

modes none;
properties arch::redundancy_scheme => Primary_Backup;
end GPS;
```

```
{implementation}
  extends
  refines type
  subcomponents
  calls
  connections
  flows
  modes
  properties
```

Some Standard Properties

Dispatch_Protocol => Periodic;

Period => 100 ms;

Compute_Deadline => value (Period);

Compute_Execution_Time => 10 ms .. 20 ms;

Compute_Entrypoint => "speed_control";

Source_Text => "waypoint.java";

Source_Code_Size => 12 KB;

Thread

Code to be
executed on
dispatch

File containing the
application code

Thread_Swap_Execution_Time => 5 us.. 10 us;

Clock_Jitter => 5 ps;

Processor

Allowed_Message_Size => 1 KB;

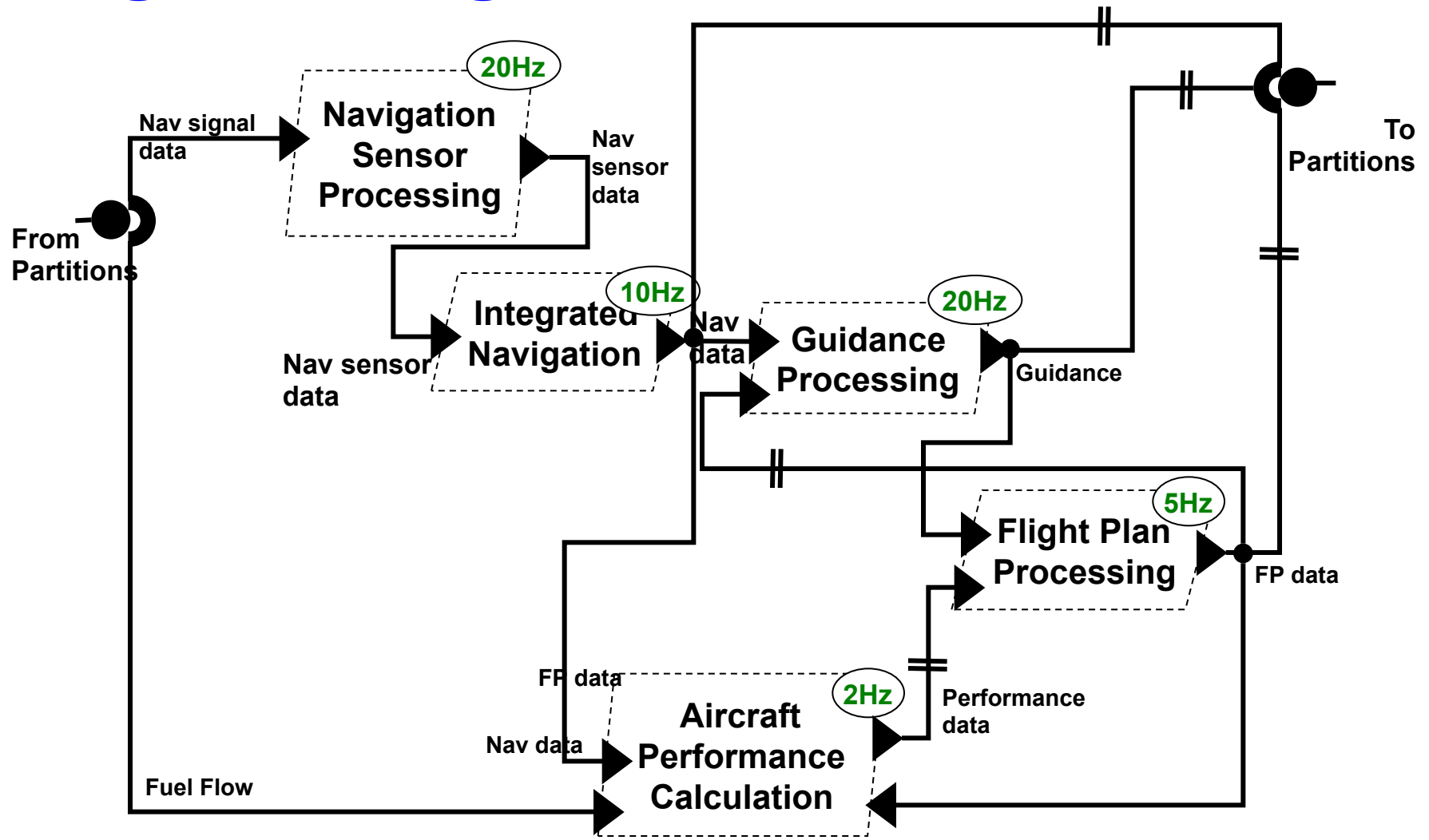
Propagation_Delay => 1ps .. 2ps;

Bus_Properties::Protocols => CSMA;

Protocols is a user
defined property

Bus

Example Graphical Specification: Flight Manager in AADL



How is AADL different from SysML?

- Think for 15 seconds...
- Let's talk...



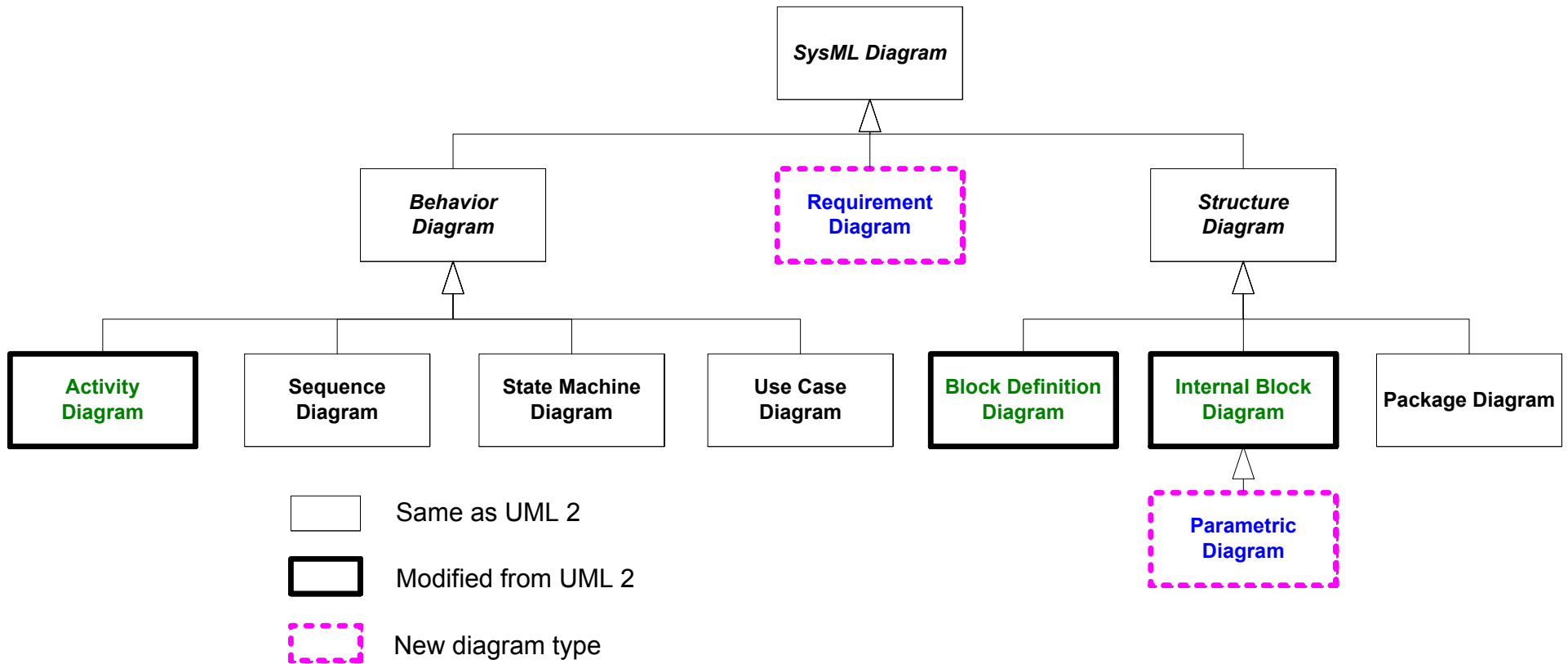


What is SysML?

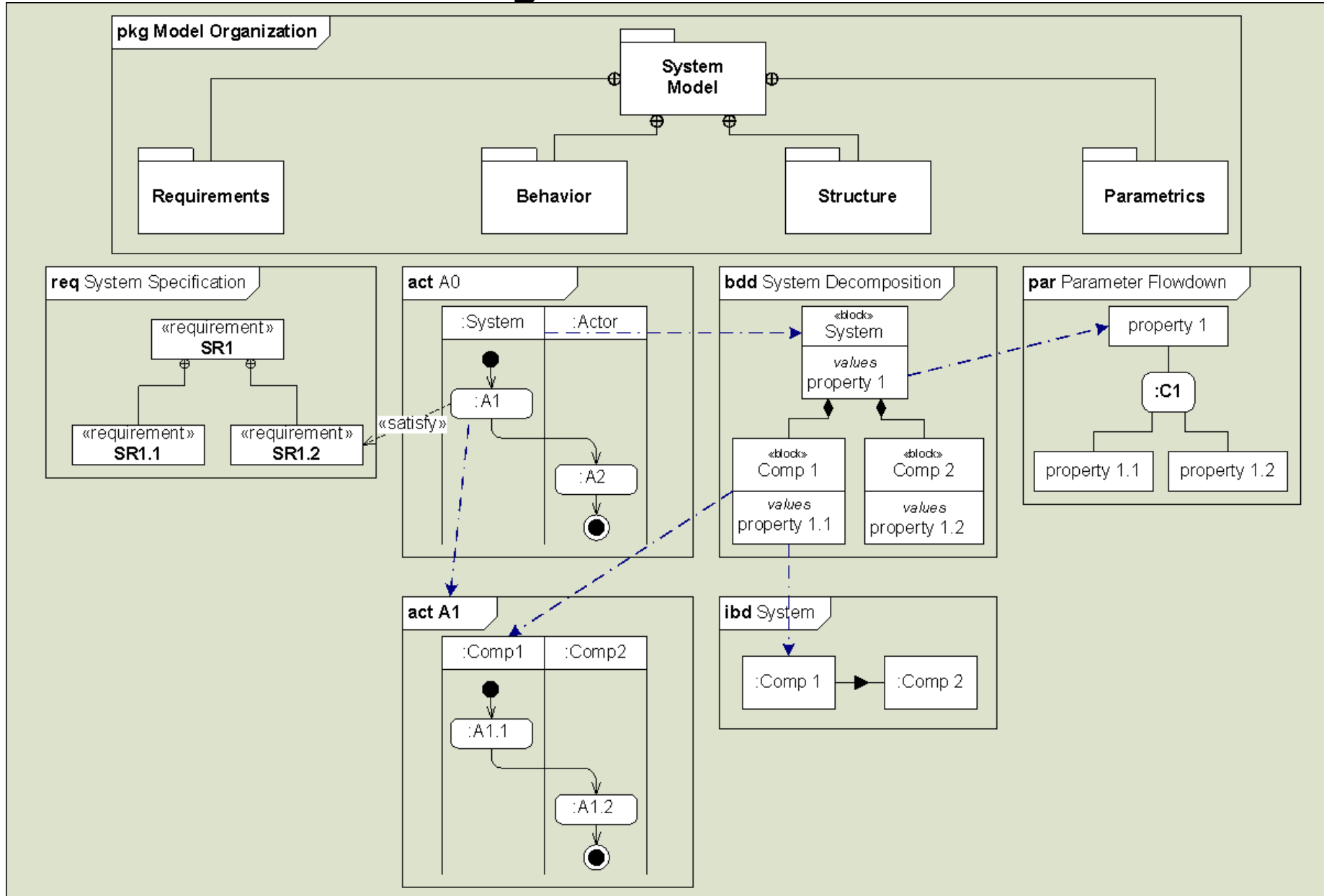


- **Graphical Modelling Language for Systems Engineering**
 - UML Profile is a subset of UML 2.0 with extensions
 - Originating with OMG, INCOSE, and AP233
- **Supports the analysis, specification, design, verification, and validation of systems**
 - HW, SW, data, personnel, procedures, & facilities
- **Supports model and data interchange via XML Metadata Interchange (XMI)**

SysML Diagram Taxonomy

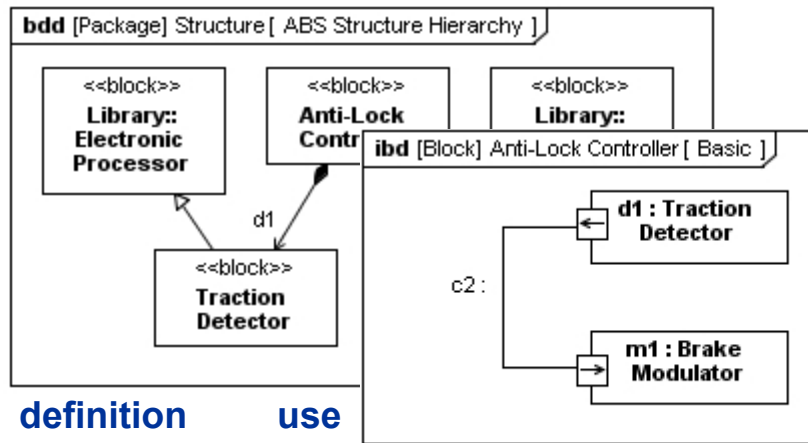


4 Pillars of SysML



SysML – ABS Example

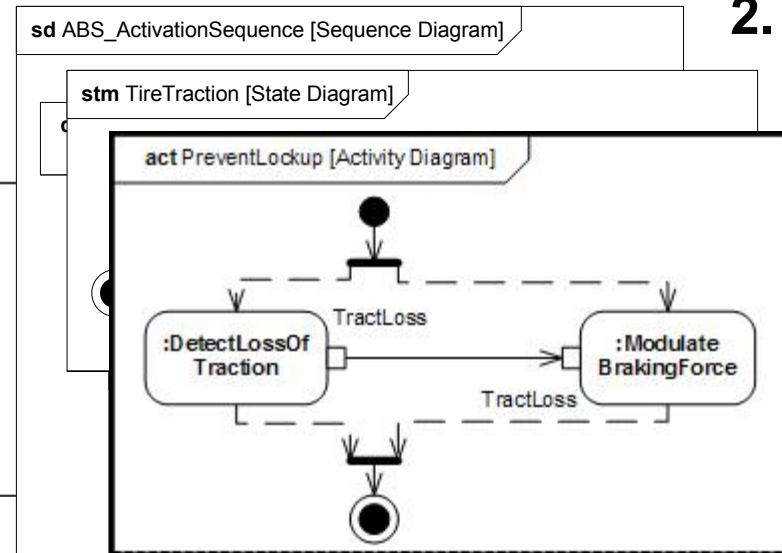
1. Structure



definition

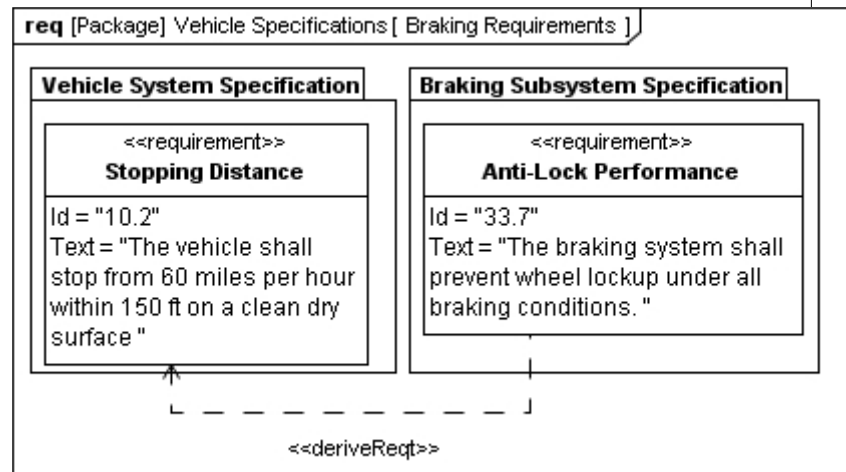
use

2. Behavior

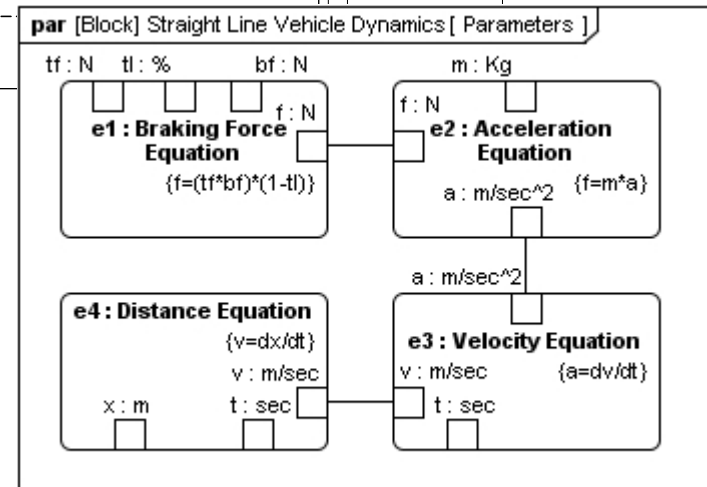


interaction
state
machine

activity/
function

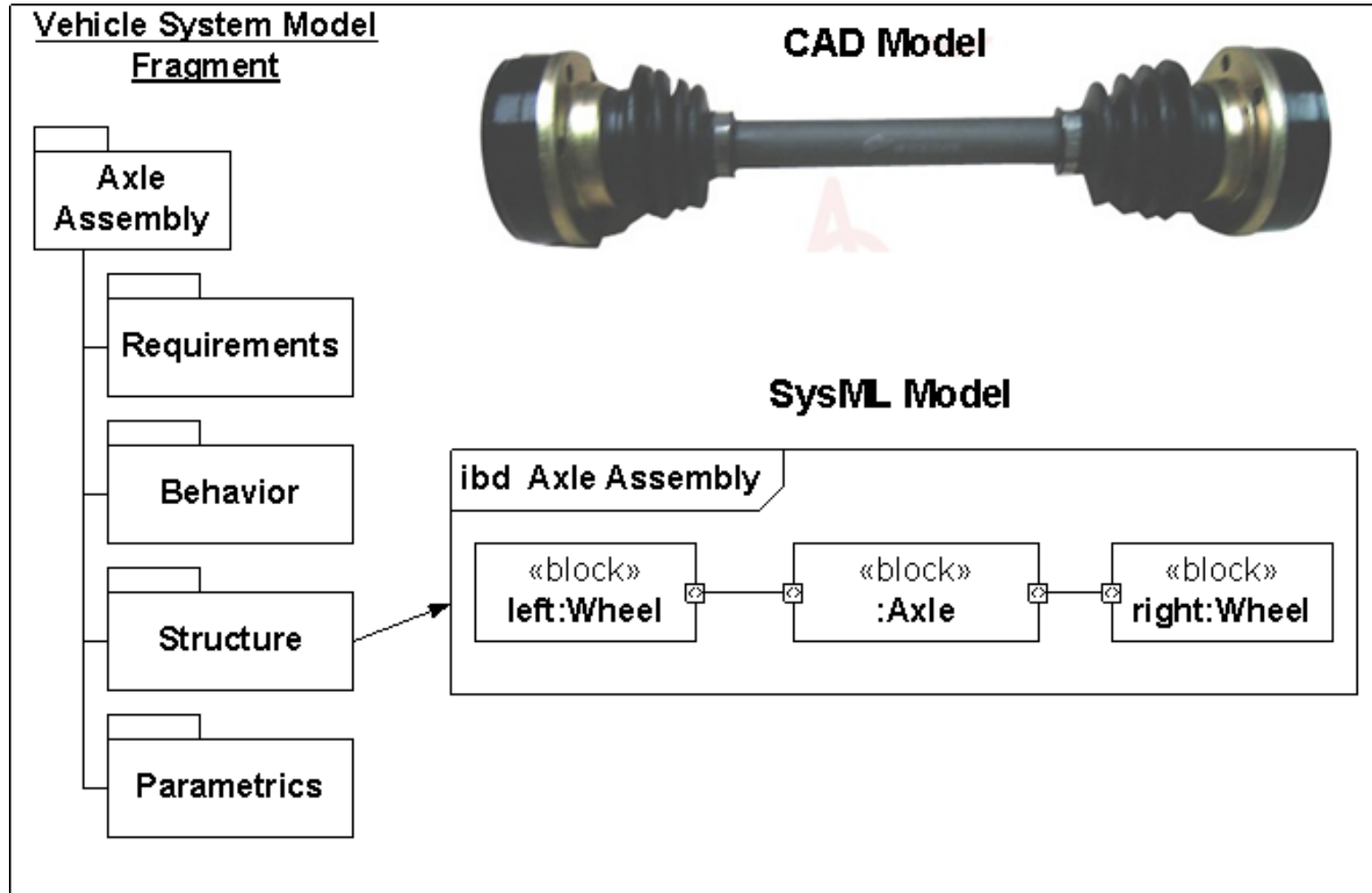


3. Requirements

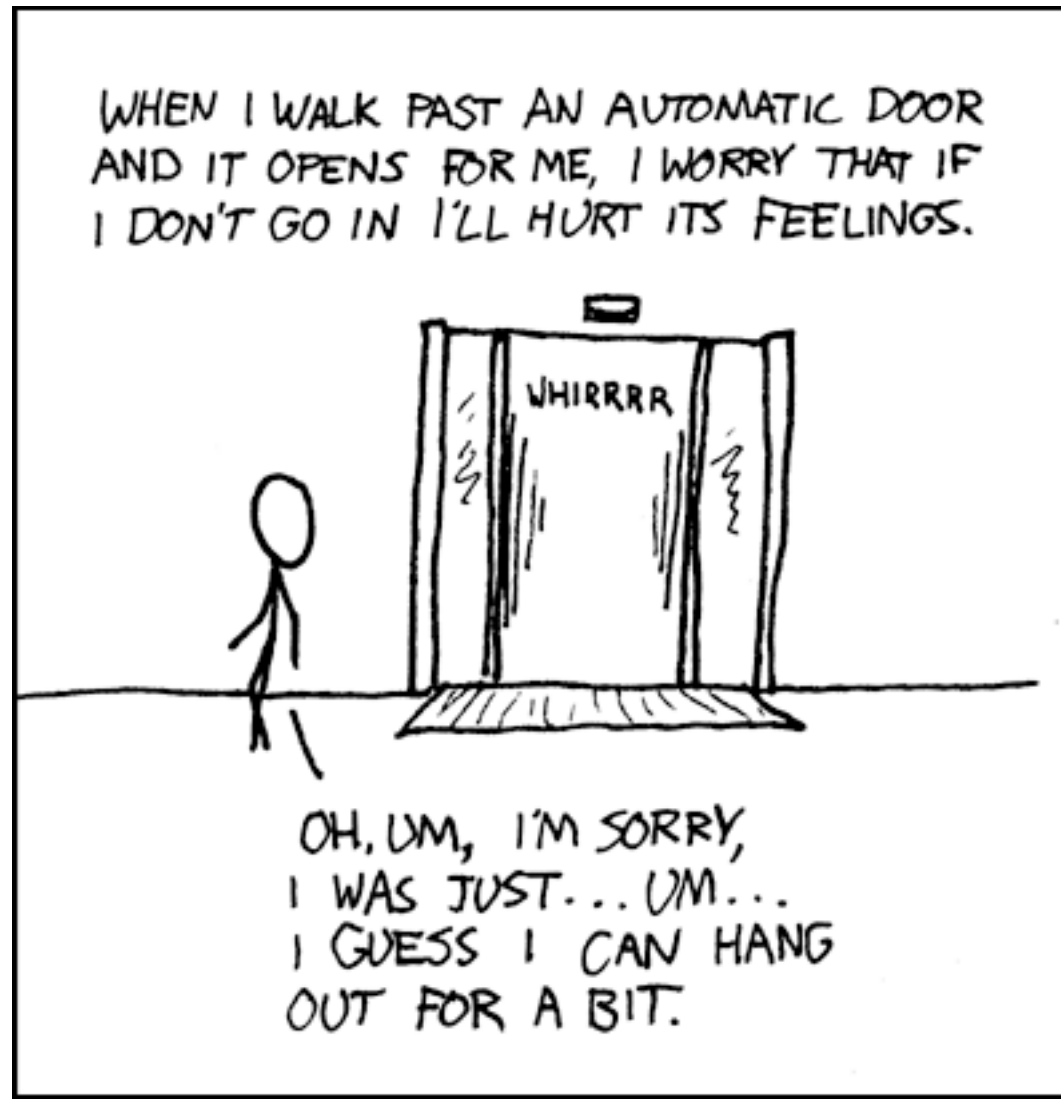


4. Parametrics

SysML Model Used to Elaborate System and Component Requirements



What humanoid robots will be like?





Homework and Milestone Reminders

- **Read Chapter 11 in text**
- **Milestone 3: Light-Weight Transformation Environment (see Milestone 3 assignment)**
 - **Due by 11:55pm, Tuesday, May 3rd, 2011.**