

# Improving Project Management Using Formal Models and Architectures

Ian Sturken

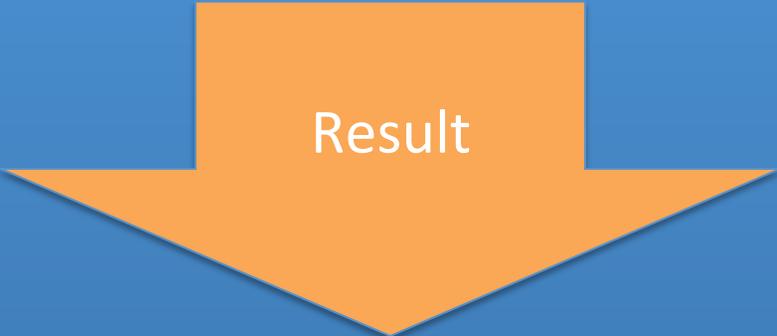
Ted Kahn

Project Management Challenge

February 9-10, 2011

# Problem Statement

Today, project information is stored in unstructured documents



Result

- Information that is inconsistent
- Information that has no traceability

# Objectives

You will have a general understanding and knowledge of:

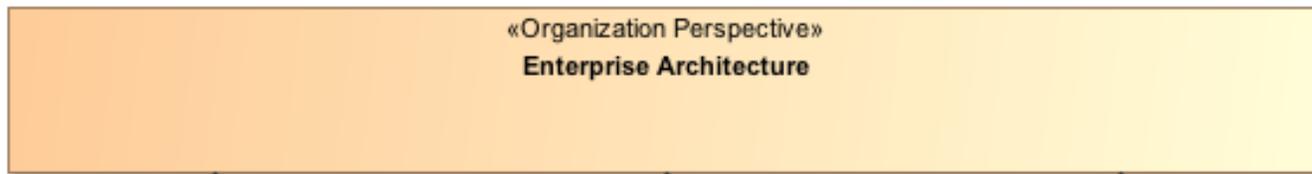
- A theoretical background of enterprise architecture, business architecture and formal modeling
- Actual projects using formal modeling and enterprise architectures
- The application of formal modeling to Project Management
- Specific do's and don'ts for making formal modeling work for you today: short term ROI

# Agenda

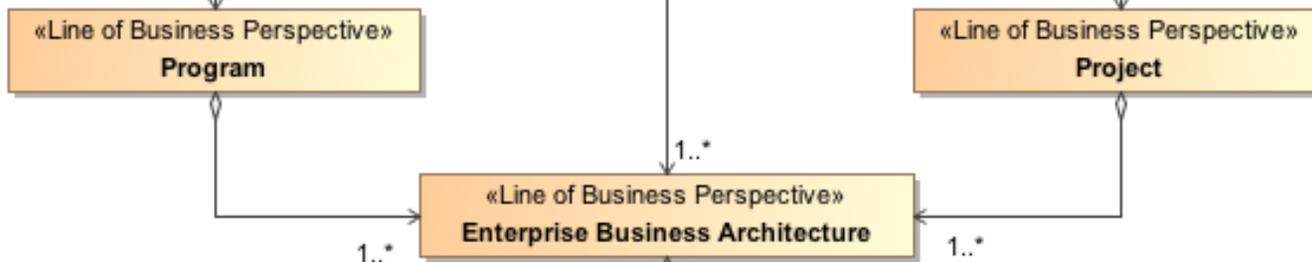
- Problem statement, objectives, agenda
- Theory of:
  - Enterprise and Business Architecture
  - Formal modeling
- Case studies:
  - Ares development
  - Ames process modeling
  - MODEAR
  - Flight Readiness System
- Applying Architecture Frameworks and Modeling Languages to Project Management
- **Making Modeling Work for You Today**
- Future Trends and Closing Remarks
- Q&A

# Four Modeling Perspectives

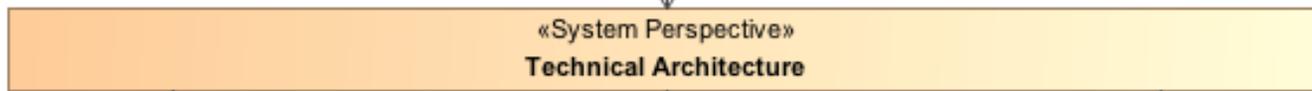
1



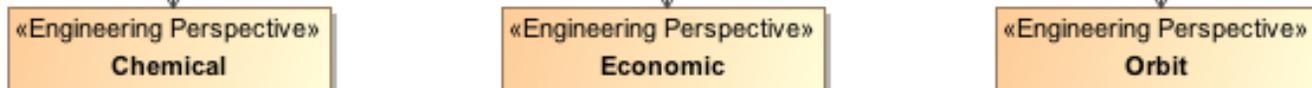
2



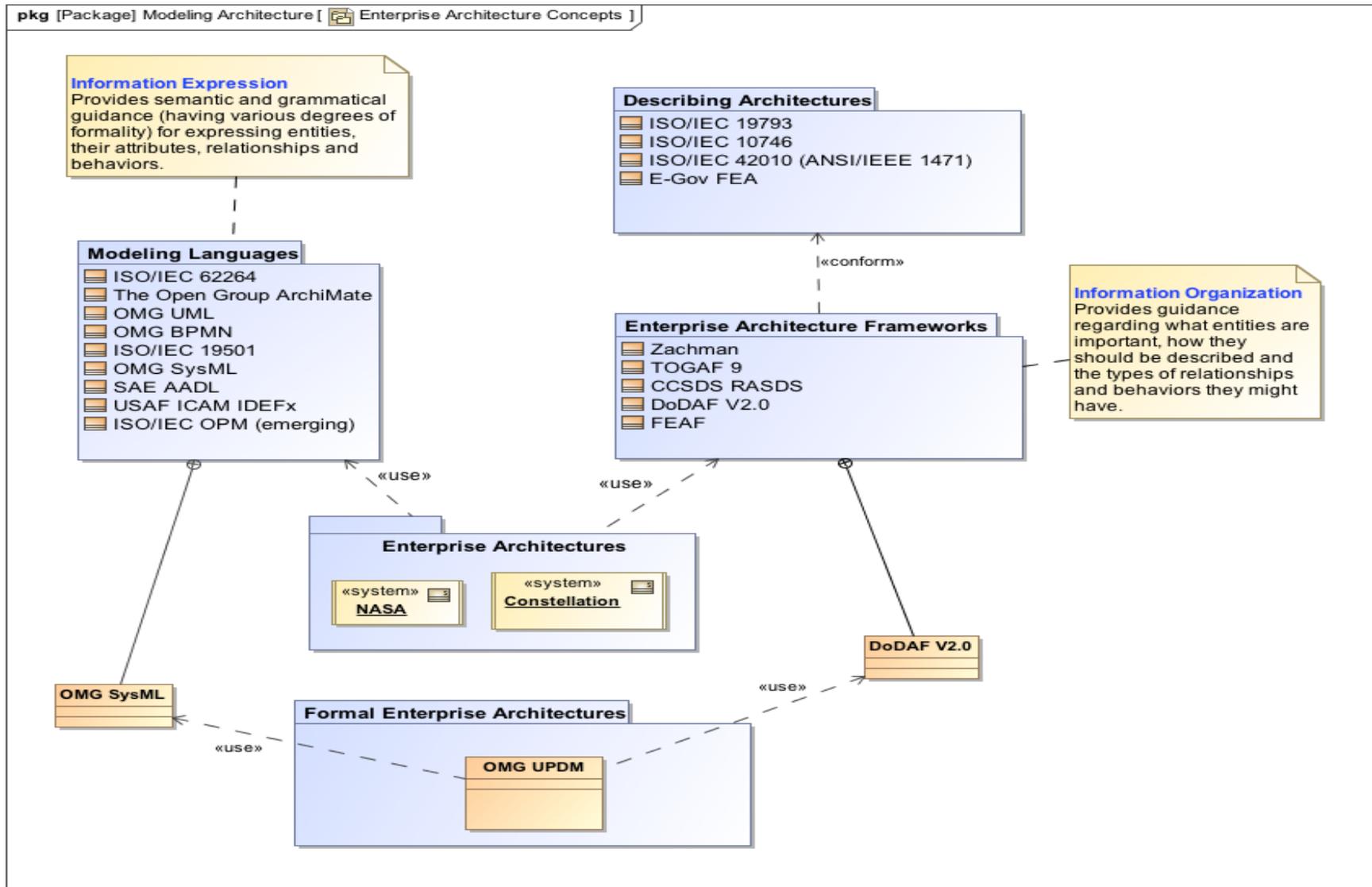
3



4



# SE Standards, Languages, AFs



# **ENTERPRISE ARCHITECTURE**

# What is an Enterprise Architecture?

## An Abstraction of the Physical World Around Us

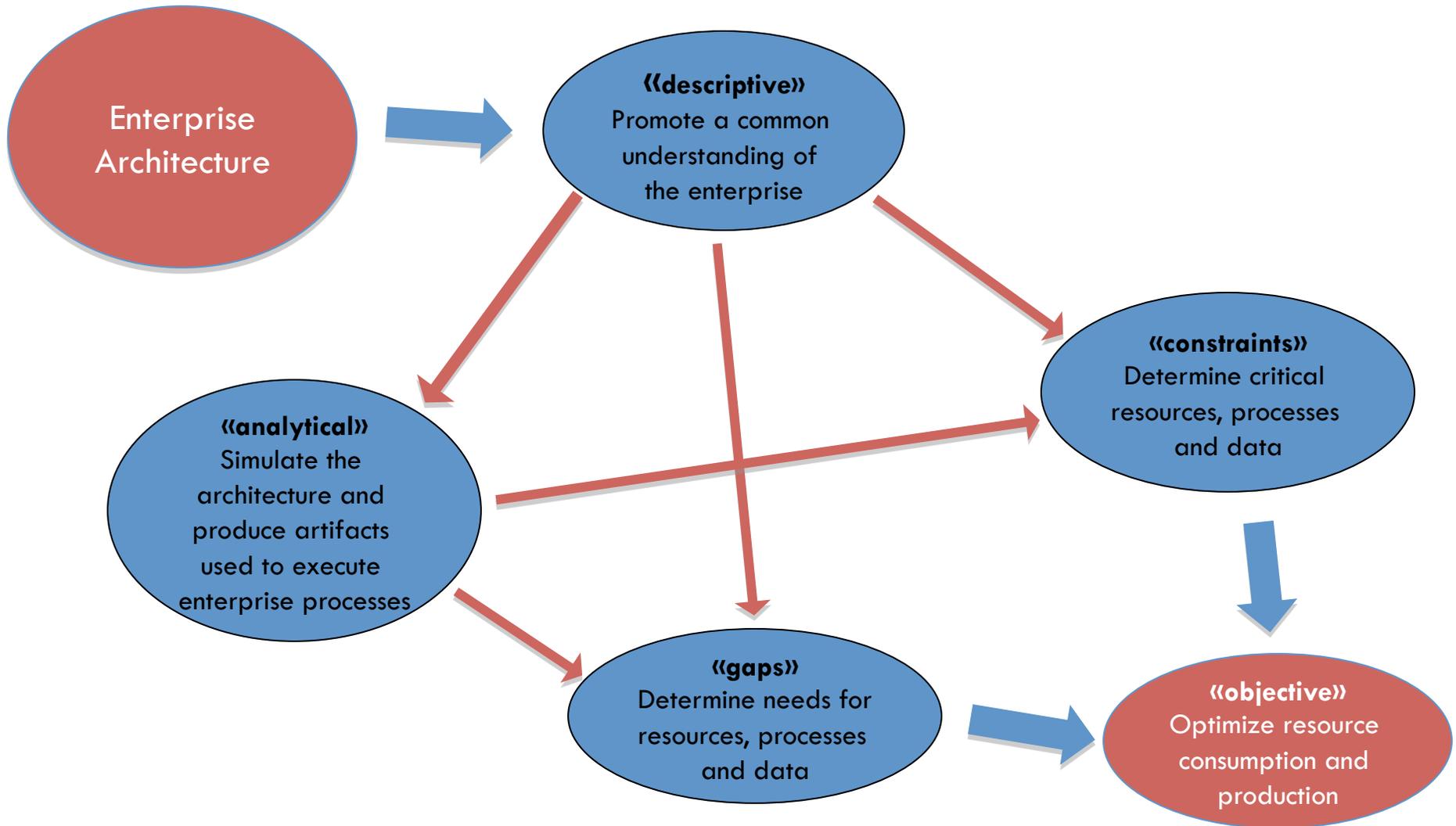
- An accounting of an organization's **IT artifacts** and their application to lines of business. (**Lists** of IT things.)
- The relationships and behaviors of an organization's **IT artifacts** and their application to lines of business. (**Lists and Life-cycle** of IT things.)
- An accounting of an organization's **meaningful artifacts** and their application to lines of business. (**Lists** of things.)
- The relationships and behaviors of an organization's **meaningful artifacts** and their application to lines of business. (**Lists and Life-cycle** of things.)

Are DoDAF, FEA, and Zachman Enterprise Architectures?

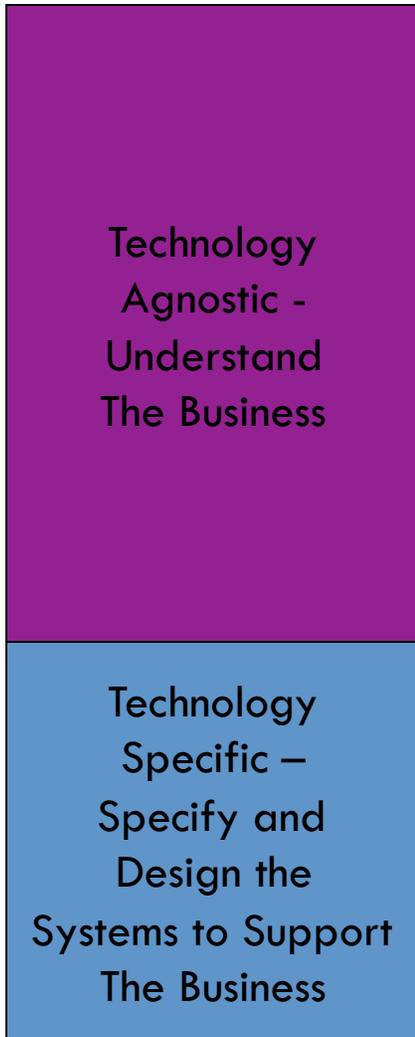
# Enterprise Architectures....

- Enterprise architectures prescribe what entities or “things” in a system or organization are important, how the information about these entities should be characterized and the relations they should have among themselves
- Systems can be an enterprise, organization or system of systems
- Entities can be people, processes, products, resources , etc.

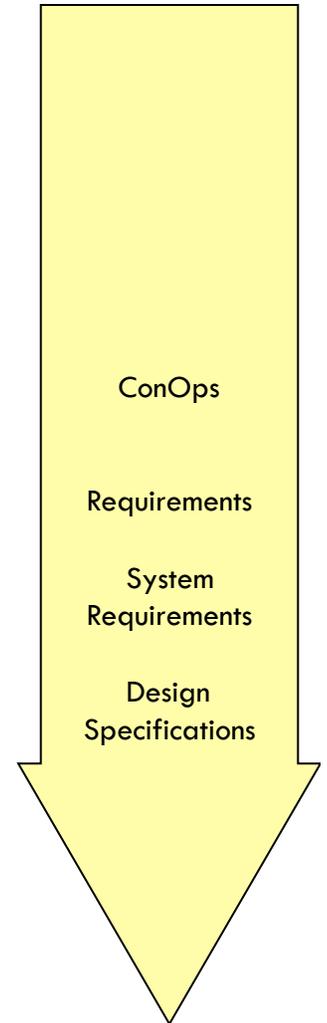
# Benefits of Enterprise Architecture



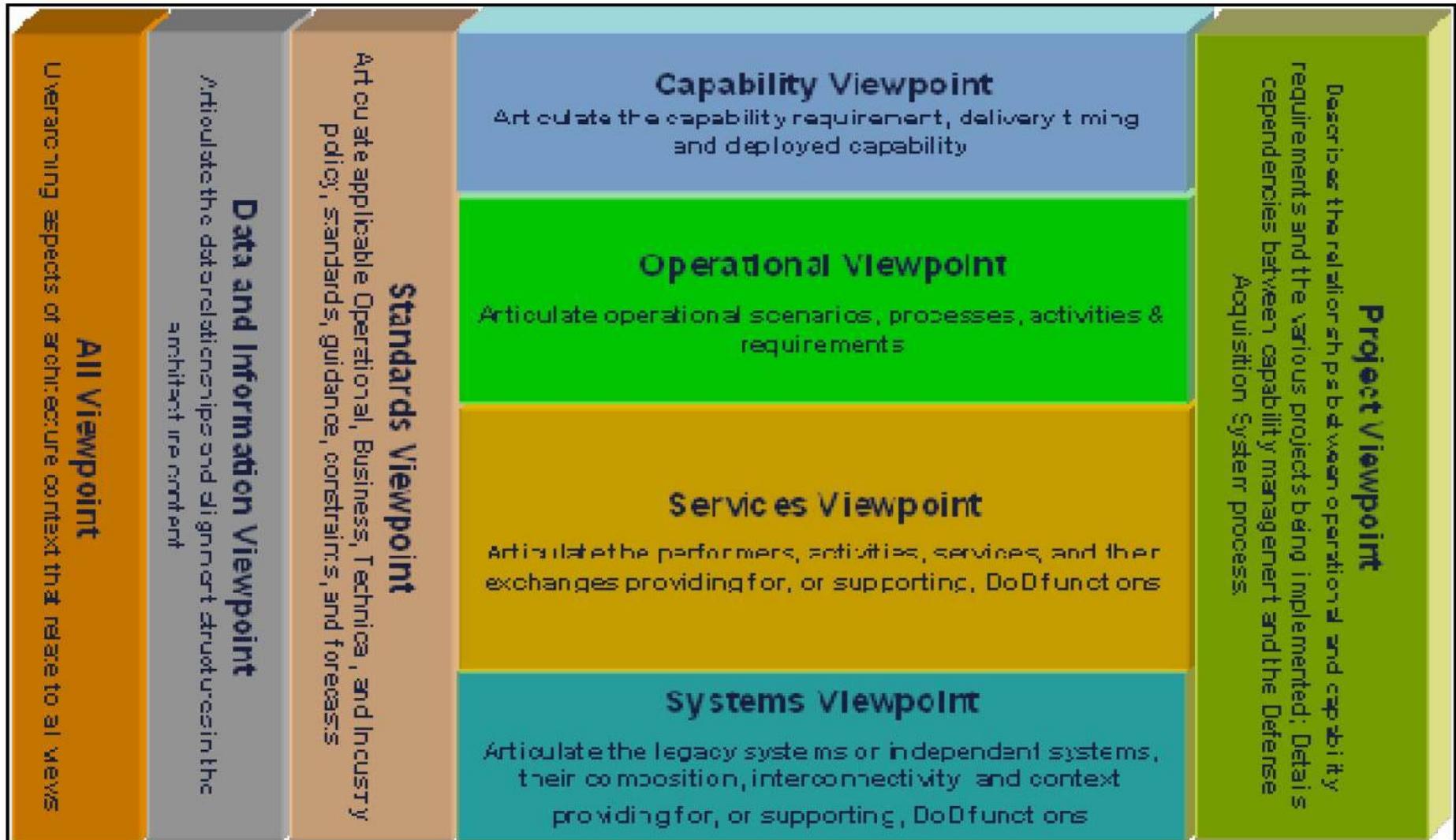
# Zachman Framework



	What (Data)	How (Function or Process)	Who (People)
Scope	•List of things important to business	•Function Hierarchy •Functions to Org Matrix	List of Organizations
Business Model	•Conceptual Data Model	•Process Model	•Org to Function mapping (roles)
System Model	•Logical Data Model	•Use Case •Activity Diagram	•Process to Role Matrix
Technology Model	•Physical Data Model	•Activity Diagram •Sequence Diagram	•Roles/Access Matrix
Detailed Design	•Technology Specific	•Technology Specific	•Technology Specific



# DoDAF Framework



**DoDAF Viewpoints**

# Which AF do I Use?

## ▪Zachman

- Easier to grasp and get started with. Can start with lists of “things” and start relating these to other parts of the business
- Hierarchical in nature, provides good mechanism for abstracting levels of detail from executive to engineer
- More IT centric

## ▪DoDAF

- More prescriptive in nature – specific products to fill different purposes
- Separate different viewpoints – business processes from systems that support them
- Supported by many tools
- Has a modeling language specifically designed for it:  
UPDM
- General purpose

# Architecture Framework, Model or Both?

- Architecture Frameworks:
  - Can range from simple (lists) to complex
  - Useful for providing an outline of what information to gather and how to organize that information
  - Can customize this outline to fit your needs
  - Can be used to compare different systems from different vendors
  - Can be used to study “as-is” states to “to-be” states
  - Can leverage modeling languages such as SYSML, IDEF, UML, UPDM
- Modeling Standards
  - Can range from simple to complex
  - Quick to build a few diagrams
  - For larger projects will need to organize model
  - Formal language/annotations used
- Both
  - Provides guidance on what modeling artifacts you will need and how to organize them according to a standard framework

Models, Formal Models and SysML

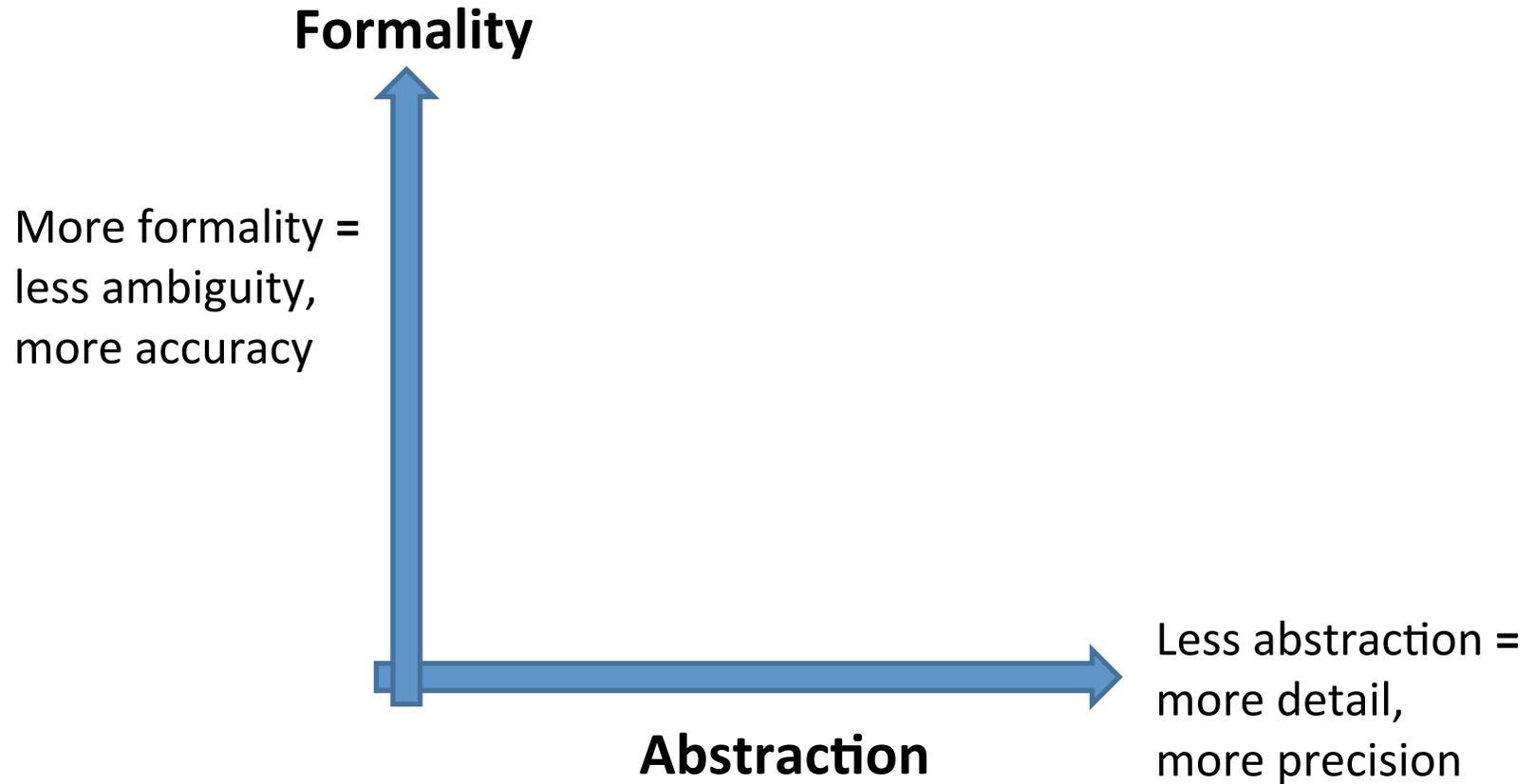
# **MODELING**

# What is a Model?

## An Abstraction of the Physical World Around Us

- An electrical schematic of a radio
- An economic model
- A mathematical model
- A model student
- A non working model airplane
- A written description of a pencil
- A diagram
- A spreadsheet
- Music
- Art
- Natural languages

# Model Attributes



# Abstraction Levels

**La Joconde**



**Femme au Chapeau Orné**



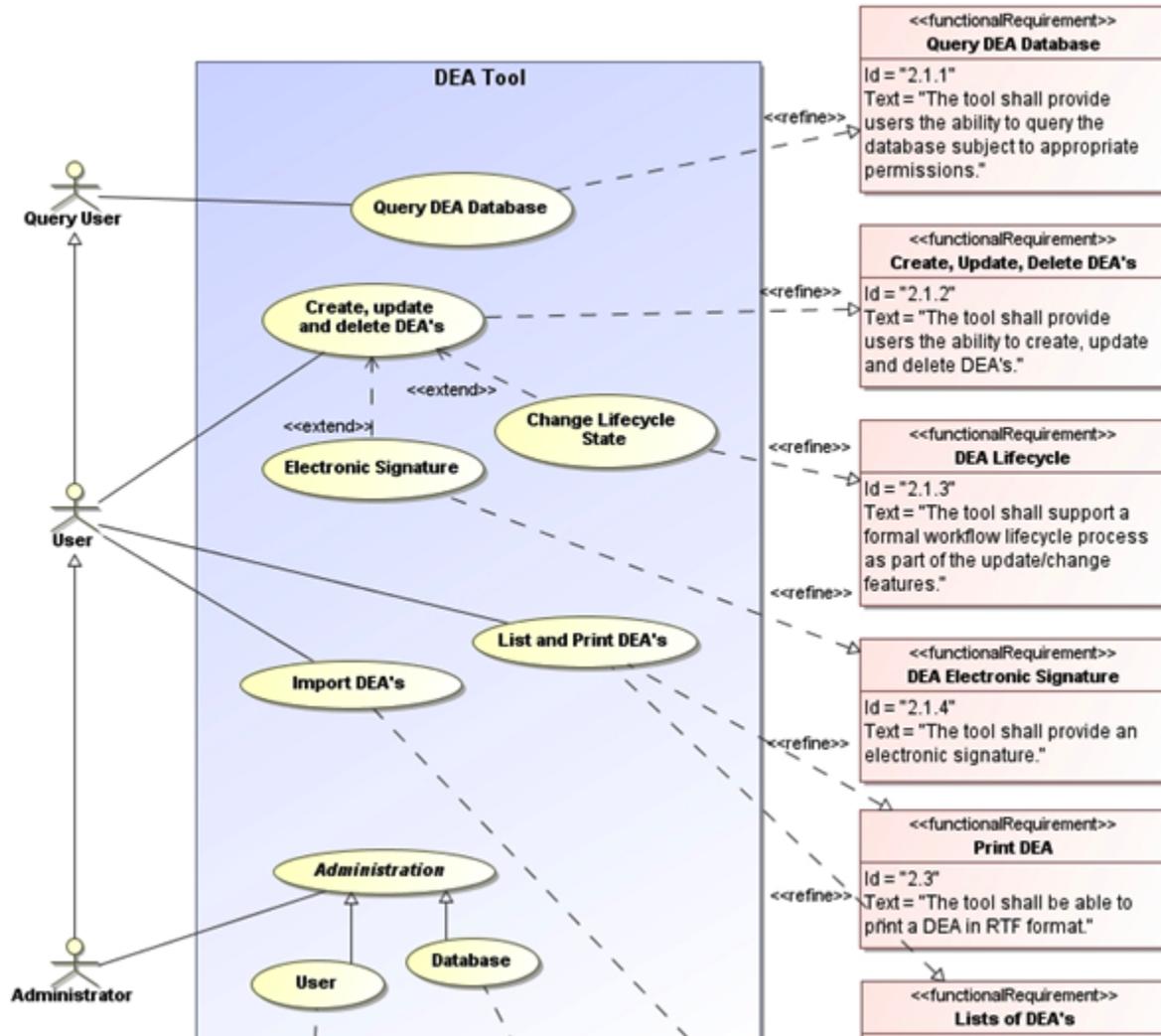
# What is a Formal Model?

The degree to which the model adheres to:

- Well defined **semantics**: model components have precise interpretations.
- Well defined **grammar**: model components can only be connected using precise structural rules.

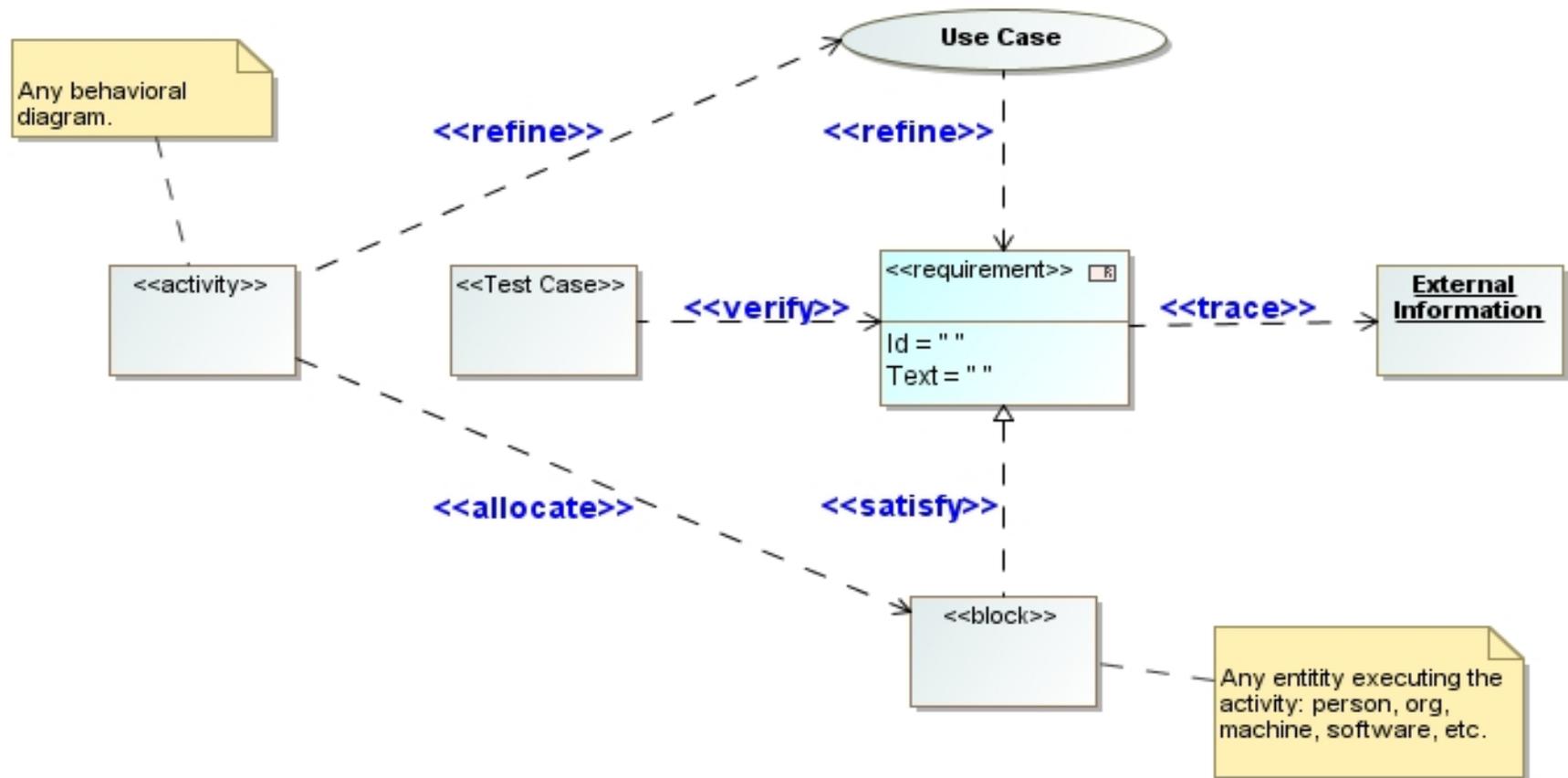
# SysML Semantics

DEA Tool Functional Use Cases

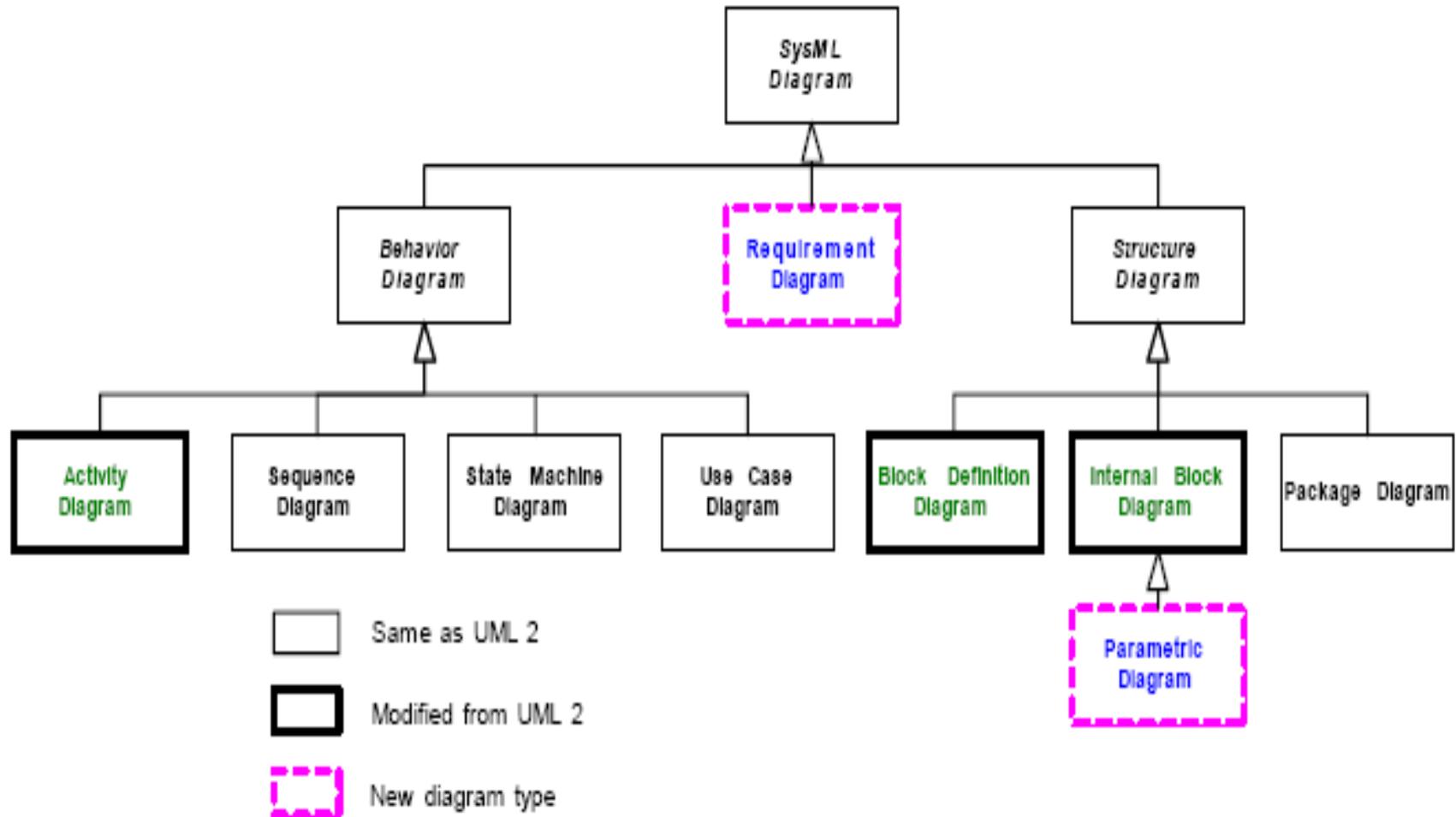


# SysML Requirements Relationships

## SysML Requirements Dependency Relationships



# SysML Diagram Taxonomy



# **CASE STUDIES**

# MOD Flight Production Process Re-engineering

- Goal: MOD needs to transform into an agile organization to be able to quickly meet needs and opportunities that arise in the next decade.
- Challenge: Currently, most information about how we conduct business is housed in different documents, spreadsheets, systems and other repositories. It is difficult to gain a comprehensive, integrated, common view of the way we conduct business and what the impact of changes are on our people, processes and systems.
- Approach: An enterprise architecture provides a framework that will allow us to organize information about our people, processes and systems in an organized, structured and integrated manner. This will allow us to:
  - Define, develop, validate and execute our missions with a common understanding of how our people, processes and systems will interact with one another
  - Run models and simulations of our business to validate our processes and systems and find areas where efficiencies can be achieved
  - Determine our net-readiness and interoperability capabilities
  - Find gaps and overlaps between our operations needs and system capabilities
- Benefits: An organization that can quickly assess the impact of external events saving \$\$\$\$ and reducing risk.

# MOD EA Repository



## MODEAR



- Op Nodes
- Products
- Info Exchanges
- Functions
- Op Activities
- Processes**
- Exchanges
- Tools
- Edit
- Export
- History

### Browse Process Instances

Filter Number/OpNode/Title:  [Clear filter](#)

Op Node	Number	Process Instance Title
<a href="#">DA6</a>	2.4.2_1	Integrate Flight Software Development and Issues Resolution - Flight Design Post MRD Instance
<a href="#">DA6</a>	2.4.2_2	Integrate Flight Software Development and Issues Resolution - Flight Design Uplink Instance
<a href="#">DA6</a>	2.4.2_3	Integrate Flight Software Development and Issues Resolution - Flight Design DOLILU Instance
<a href="#">DA6</a>	2.4.3_1	Provide Flight Software Change Coordination - Instance 1

#### Source Process Instances (and exchanged products)

[4.2.1.2\\_S2\\_1 - Update Trajectory and I-Load Design Instance - Step 2](#)  
--> [Flight Design Flight Software Discrepancy Report \(DM\)](#)

#### Selected Process Instance

**2.4.2\_2 - Integrate Flight Software Development and Issues Resolution - Flight Design Uplink Instance**  
**Function:** [2.4.2 - Integrate Flight Software Development and Issues Resolution](#)  
**Operational Node:** [DA6 - Technical Integration and Production Control Office](#)  
**Description:**

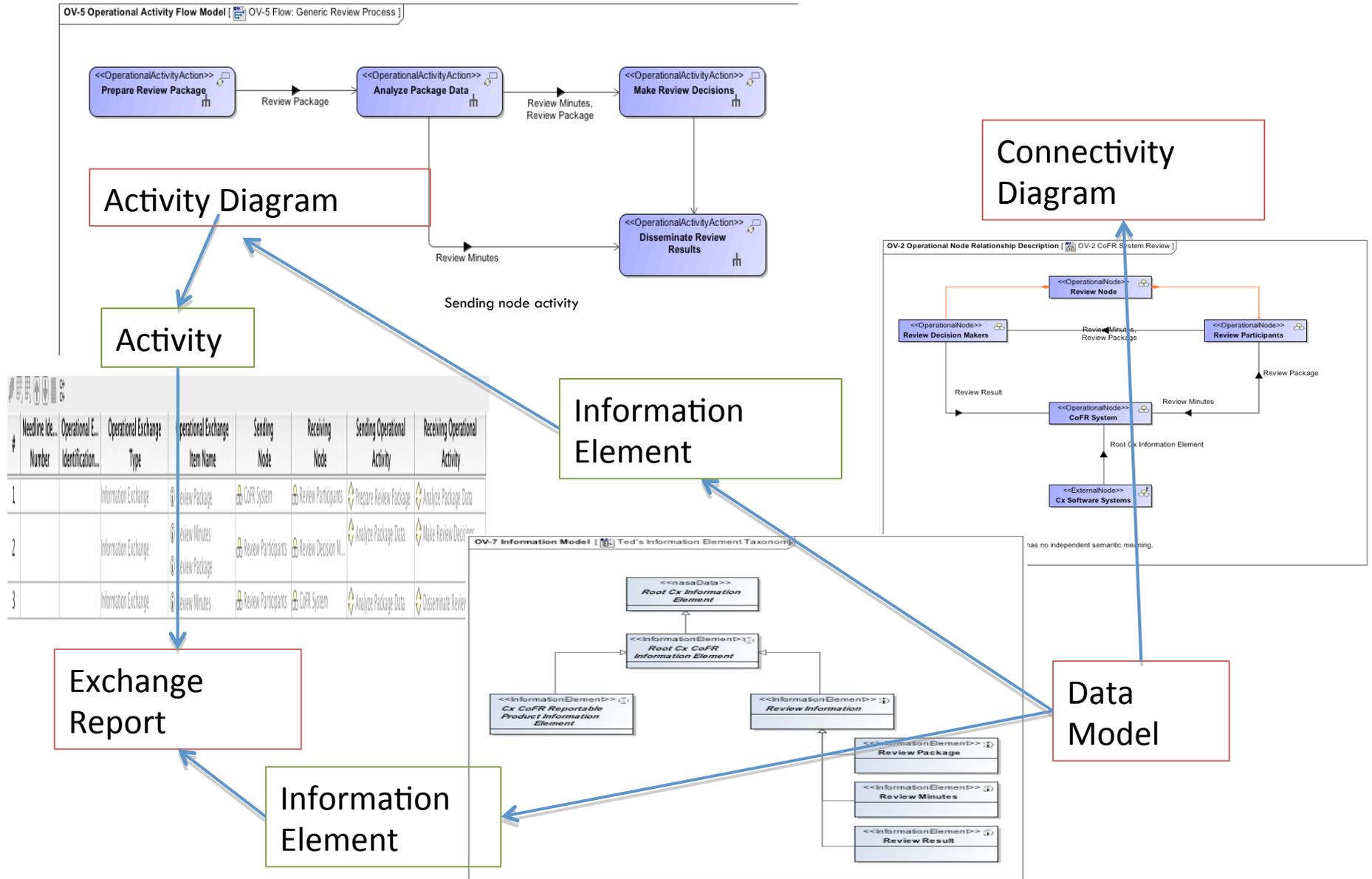
#### Recipient Process Instances (and exchanged products)

[4.2.1.3\\_S1\\_1 - Generate Day-of-Launch I-Load Update Instance - Step 1](#)  
--> [Flight Design Flight Software Discrepancy Resolution Report \(DM\)](#)

# Flight Readiness System

- Goal: Develop a new system to support certification of flight readiness for Cx
- Challenge: How do we specify the components of our system with varying levels of detail while maintaining consistency throughout
- Approach: Use DoDAF to describe the 'as-is' process and systems for shuttle. Then design a new set of processes and supporting systems for Constellation as a 'to-be'.
- Benefits: Information is organized and represented consistently with various levels of detail appropriate to different stakeholders

# Integrated Architecture

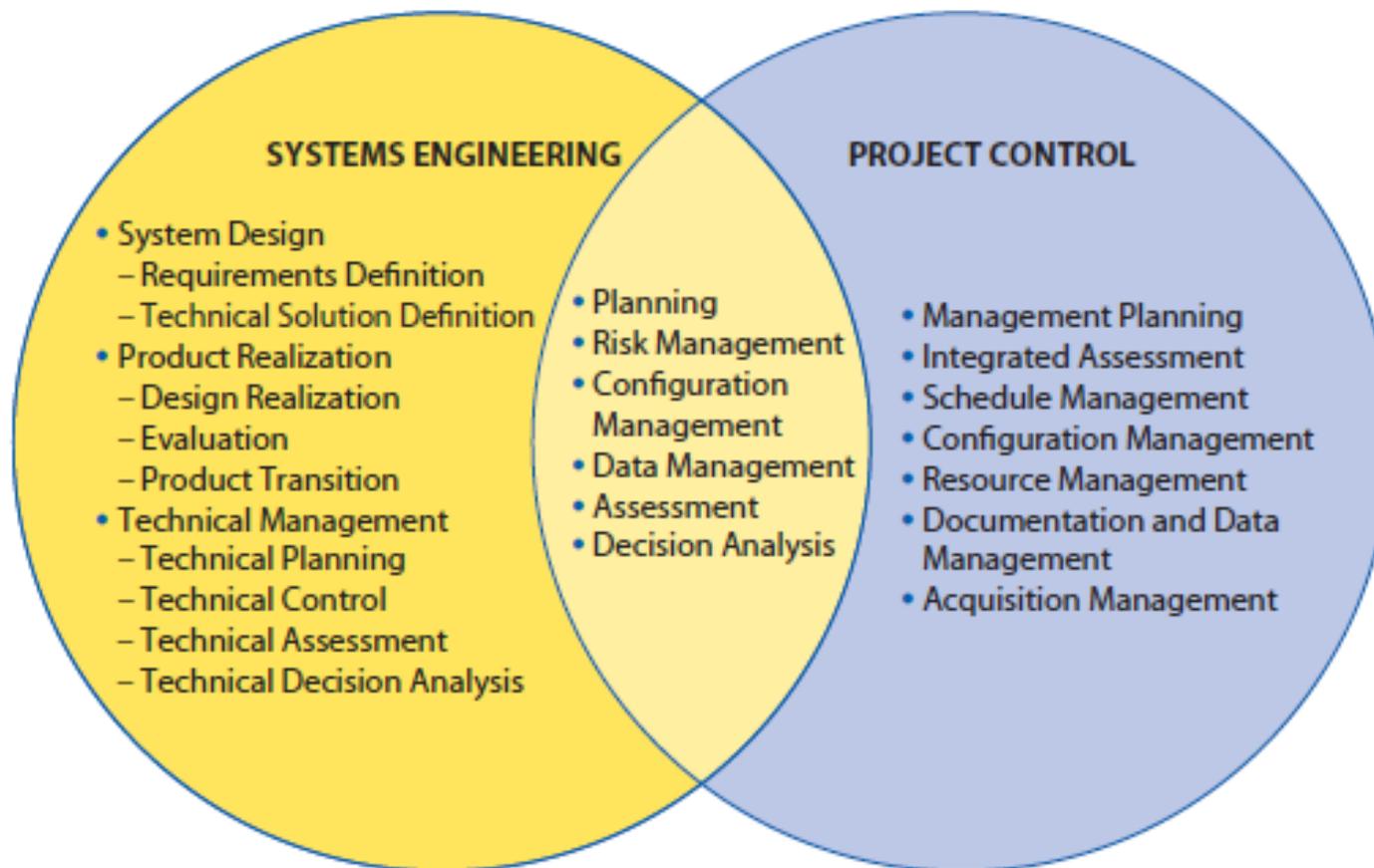


# **APPLYING EA & MODELING TO PM**

# Modeling and PM

- Projects are now modeled using spreadsheets, diagrams and documents to represent different parts (components) of the project.
- A formal model does not change this. Instead, your project components must now be represented using formal grammar and semantics.

# System Engineering and Project Management



From NASA System Engineering Handbook - NASA/SP-2007-6105

# Review Entrance Criteria

(NASA Systems Engineering Handbook)

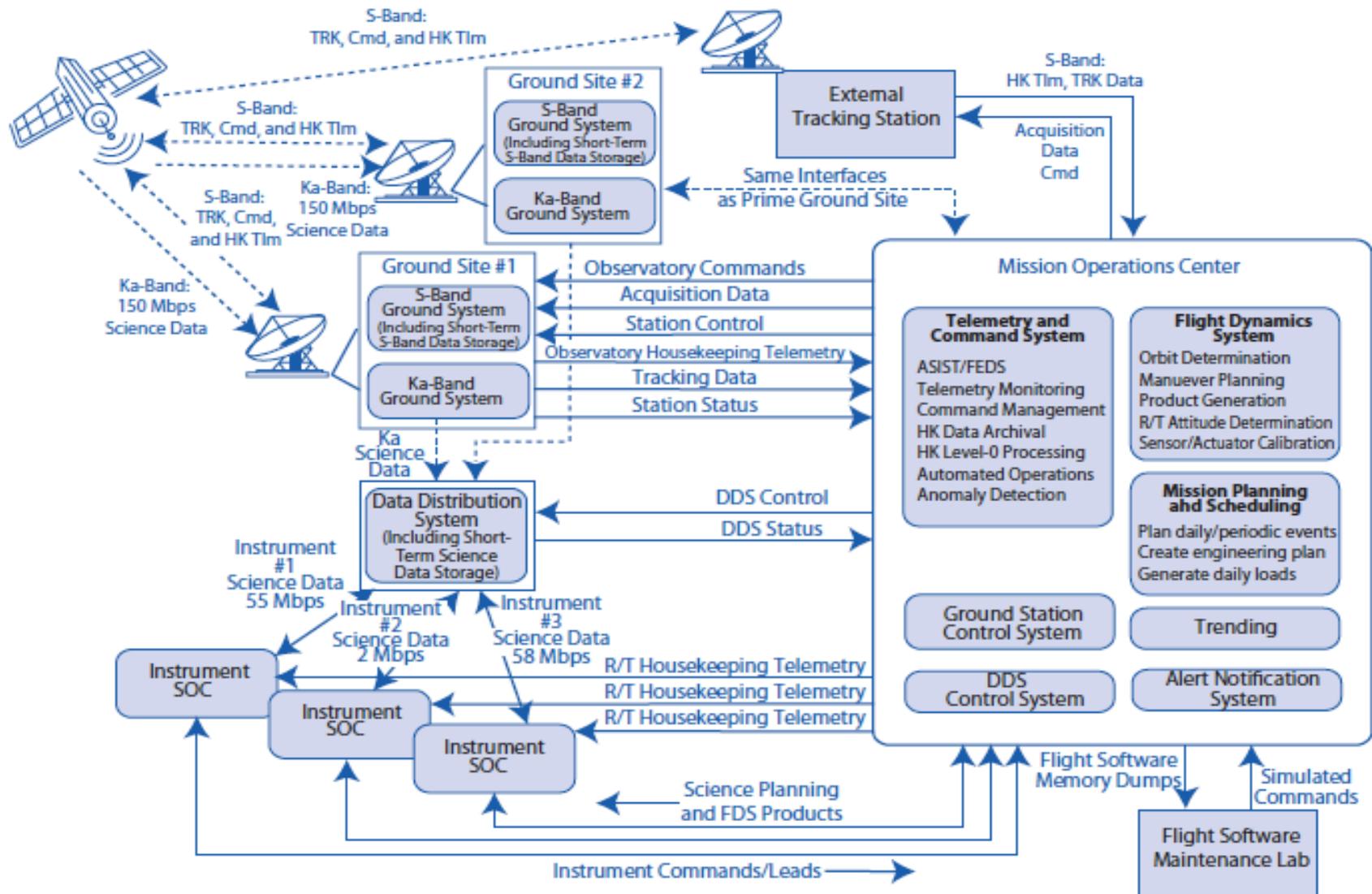
Milestone	Artifacts
System Concept Review	System Goals And Objectives
	Concept of Operations
System Requirements Review	System Requirements
	System Functionality Description
	Concept of Operations
Preliminary Design Review	Preliminary System Requirements
	Preliminary subsystem design Specs
	Operational Concept
	Interface Control Documents
	Requirements Traceability Matrix

These can all be described in one model!

# Building ConOps from Model

Conops Section	DoDAF product	SYSML Model
Scenarios	OV-5 Activity Diagram	Use Case Diagram, Activity Diagram
Conceptual Overview	OV-1 High Level Concept	Block Definition Diagram
Event sequence	OV-6c	Sequence Diagram
Connectivity Architecture	OV-2 Node Connectivity Diagram, OV-3 Information Exchanges, SV-1 System Interface, SV-2 System Communication	Block Definition Diagram
Glossary	AV-2 Integrated Dictionary	Block Definition Diagram

# An informal diagram from SE handbook – Convert to AE or SVSMMI ?

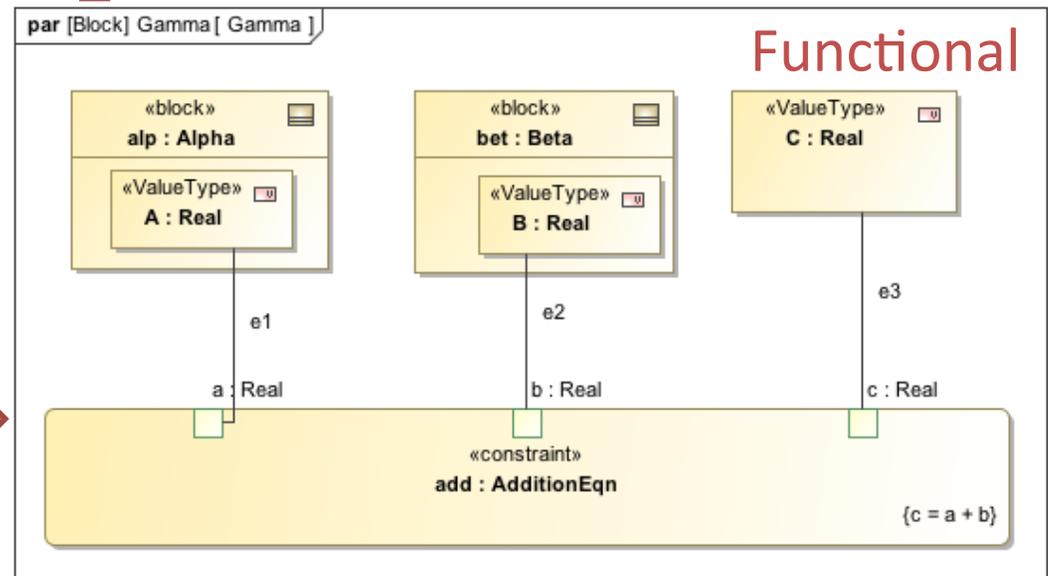
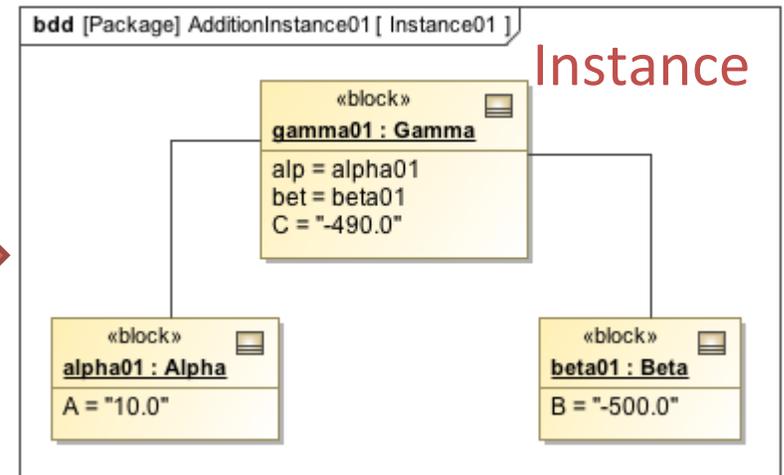
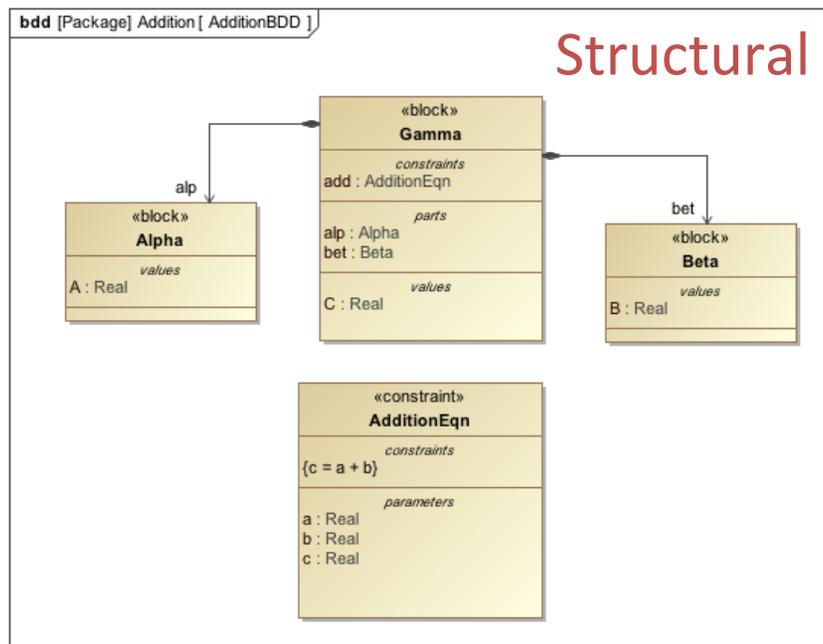


# What is the effect of changing this?

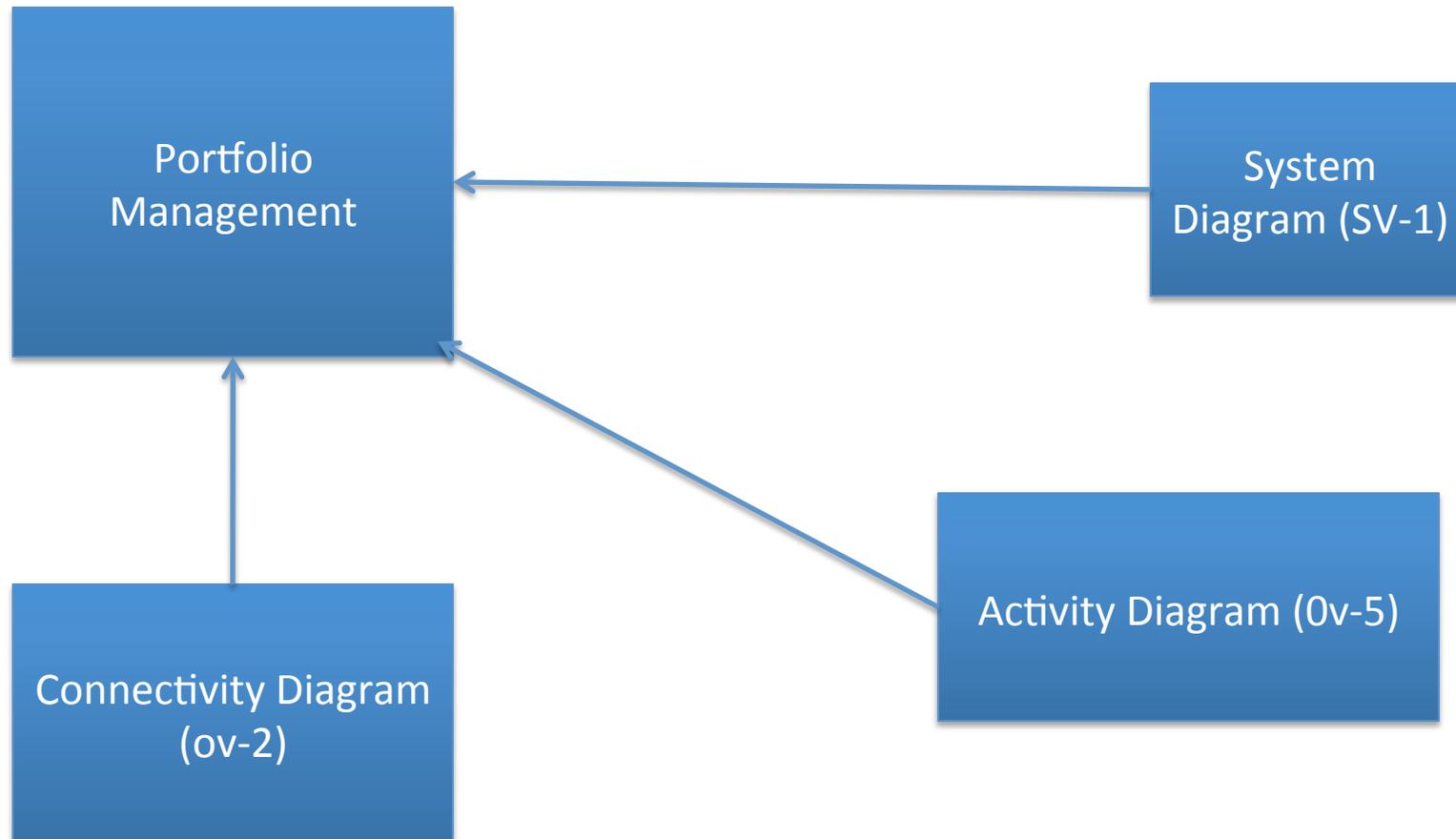
On this slide we can show that a change in one part of the model would allow you to determine what changed in other parts of the model.

# Technical Decision Analysis

## (Trade Analyses)



# Portfolio Management



# Formal Modeling and Six Sigma

(Complementary Technologies.)

	Six Sigma	Formal Models	Both Together
Methodology	Yes	No	Yes
Formal Data Semantics & Grammar	No	Yes	Yes
Data Persistence	No	Yes	Yes

# Culture Issues

- People do not necessarily want to share their information
  - Job security
  - They don't know the information, and perhaps are reluctant to say so.
  - Its time-consuming to get the information, what's in it for them?
- People like to work independently

# Modeling Summary

- Think small, think of specific questions your model should answer.
- Don't use a formal architecture.
- Do learn the semantics of your modeling language.
- Do take the time to learn your tool.
- Do pro actively manage the modeling task.

Practical Information for achieving quick ROI

# **MAKING MODELING WORK FOR YOU TODAY**

# Modeling is an Engineering Task

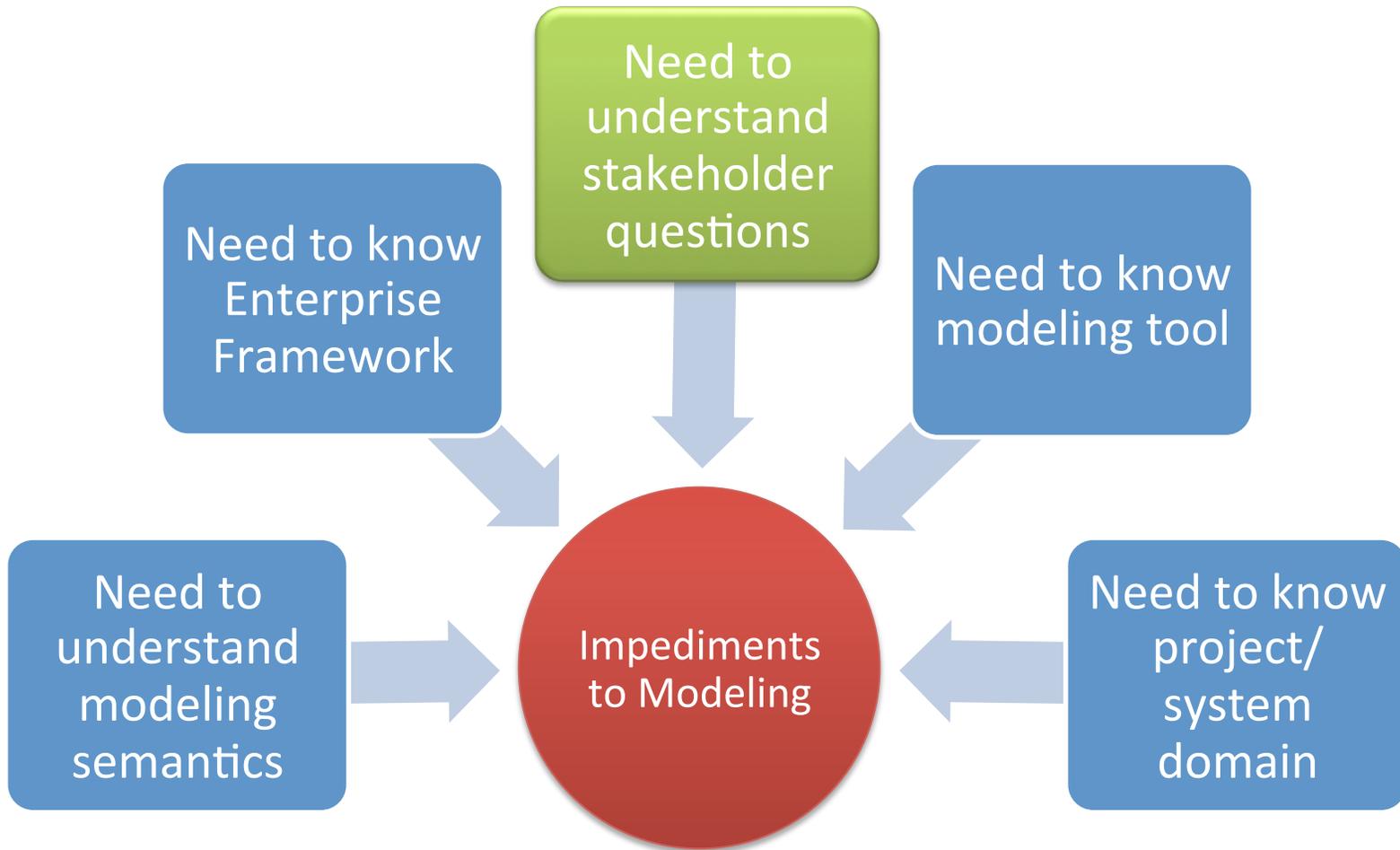
- Approach it systematically
- Know what resources you will need
- Define milestones, a roadmap
- Be pragmatic

# What Makes a Good Formal Model?

- Model those aspects of the project required to answer stakeholder questions, and no more.
- Model the degree of precision required to answer stakeholder questions, and no more.
- Models must always be accurate.

# Five Knowledge Domains

(Why modeling is hard)



# Think Small, Think Focused

(Get ROI in Weeks!)

- Think hard about stakeholders and the questions they need answered. (You are a stakeholder.)
- Determine whether you need to utilize and architecture framework to satisfy your stakeholders
- Do not get hung up on tools, useful comparisons can be very difficult to make
- Use SysML, unless you have a compelling reason not to. And learn the relevant parts.

# Modeling Tools are Hard

(In fact, they are two tools in one.)

- Database program
- Drawing program

Always make sure you know  
which action you are taking:  
database or drawing.

# Two Modeling Steps

Project knowledge

translate to

SysML knowledge

apply to

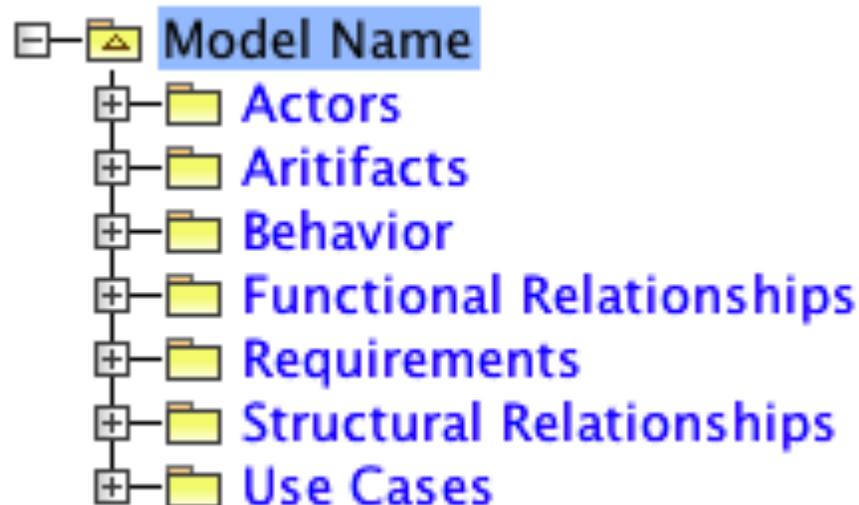
Tool knowledge

Translate or apply, never both at the same time

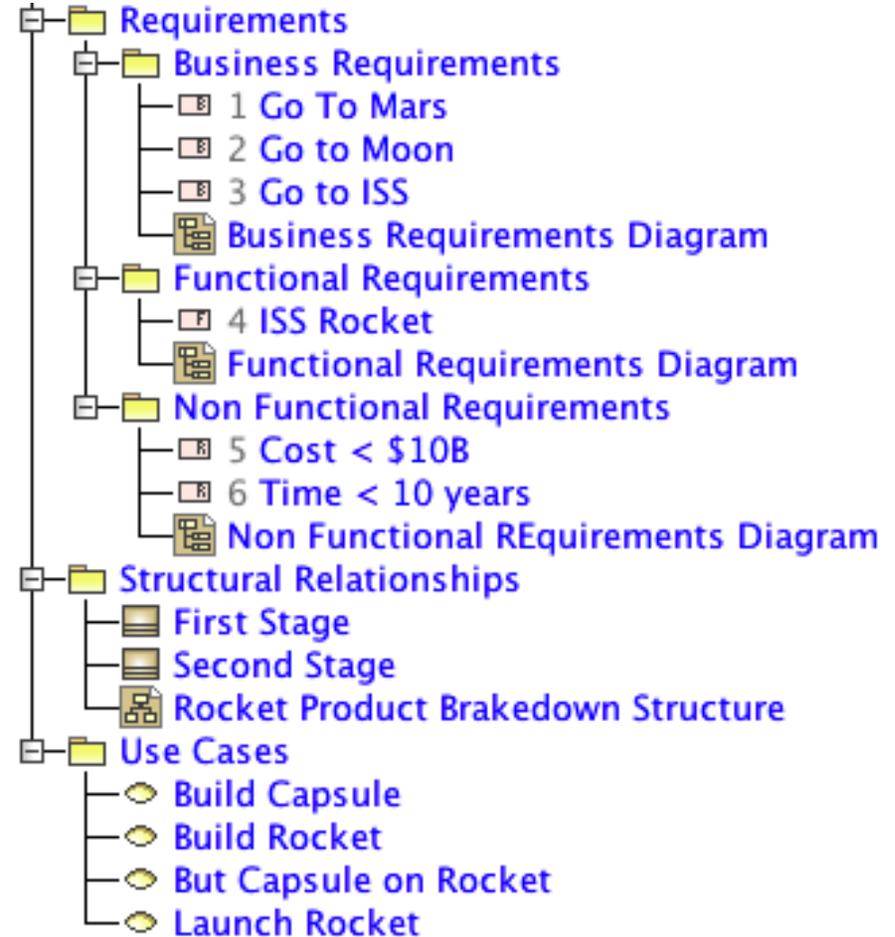
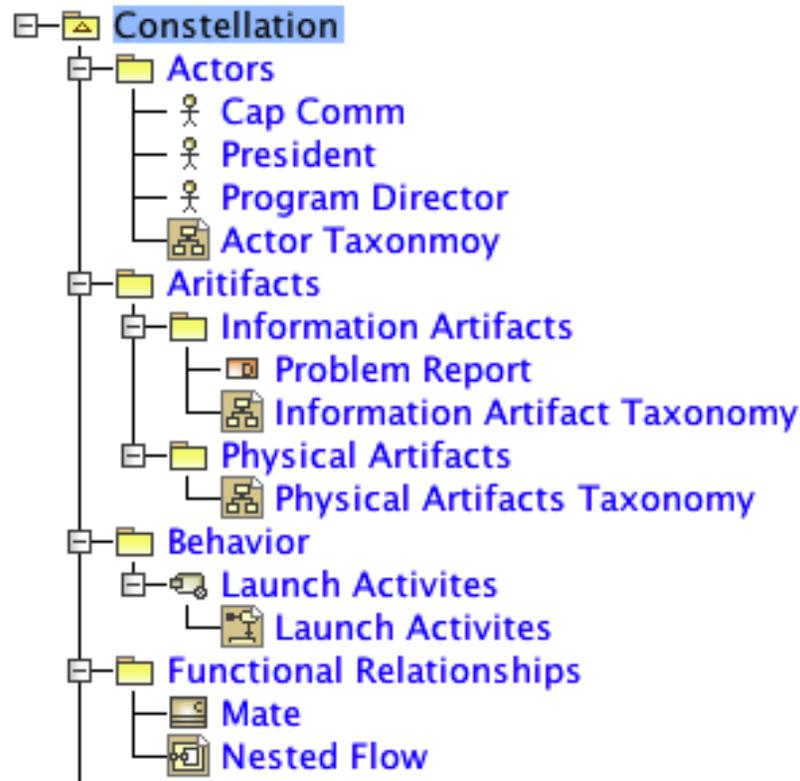
# You're in Front of your Computer

(Now what do you do?)

- Your tool is running
- You created a new SysML project
- And...
- You create packages to organize your project



# A “Template” SysML Model



# Extending SysML

- Use English to document each entity.
- Use diagram notes to highlight explain diagram elements.
- Use SysML Profiles to extend SysML semantics to meet your own domain specific needs.

# Modeling Tips

- What if you don't know something?
  - Make your best guess, its easy to change.
- What should go on a diagram?
  - It should tell a story, answer a question, address a specific stakeholder need.
- Look to see how a set of diagrams might meet a stakeholder's need in some specific area.
- Model only those elements for which you know there is a value.

# Culture Issues

(Modeling is about sharing information.)

- Some people do not necessarily want to share their information
  - Job security
  - They don't know the information, and perhaps reluctant to say so.
  - Its time-consuming to get the information, what's in it for them?
- Some people like to work independently

# Modeling Summary

- Think small, think of specific questions your model should answer.
- Don't use a formal architecture.
- Do learn the semantics of your modeling language.
- Do take the time to learn your tool.
- Do pro actively manage the modeling task.

# **FUTURE TRENDS AND CLOSING REMARKS**

# Future Trends

- Fully define semantics
- Prescriptive methodologies
- Improved tooling
- Analytical integration

Project Management Information:  
Gantt Chart, Event Chain Diagram, ISO 10006, Six Sigma

5

...producing **strategic**  
PM artifacts...

...leading to better estimates for  
schedule, scope and resources..



...and also used for aligning project  
schedule, scope and resources...

...maintaining a consistent, feasible  
project and a refined model...

4

Formal Model = Formal semantic relationships + consistent representations  
⇒ improved common understandings & decisions.

3

...used for creating  
SysML model...

...providing consistent **operational**  
information used by all stakeholders...

2

...which encompasses  
project data...

Text Docs

Spreadsheets

Diagrams

1

Represented by...

Projects are defined and change mostly due to external events: Continuously.