MODEL BASED PROJECT MANAGEMENT

Applying the Systems Modeling Language to Project Management

Theodore Kahn
Intrinsyx Technologies Corporation
NASA Ames Research Center
Moffett Field, CA
www.intrinsyx.com
theodore.e.kahn@nasa.gov
theokahn@gmail.com

NASA/Army Systems and Software Engineering Forum
May 12 & 13, 2009
Marshall space Flight Center
Problem statement

- **Project and system information is now document centric:**
  - Relies on textual, tabular, and graphic documents from a variety of sources which means…
  - interpretation of the information is difficult and consistency hard to maintain, especially under changing conditions.

- **Model centric approach relies far less on documents:**
  - Depends instead on a single repository for storing project structural and behavioral information…
  - Promotes a more common interpretation and consistency of project information, even under very dynamic conditions.
Presentation objectives

- Provide a conceptual understanding of modeling as it might be applied to project management and system engineering.

- Enable you to make informed decisions regarding its applicability to your situation.

The talk does not address implementation issues or methodology, though those would be interesting areas for the discussion at the end.
Some basic SysML semantics
Theoretical aspects of modeling
Demonstration of a SysML tool and models
Q & A, and discussion
DEA Tool Functional Use Cases

1. **Query DEA Database**
   - Text: "The tool shall provide users the ability to query the database subject to appropriate permissions.

2. **Create, Update, Delete DEA's**
   - Text: "The tool shall provide users the ability to create, update and delete DEA's.

3. **Electronic Signature**
   - Text: "The tool shall provide an electronic signature.

4. **List and Print DEA's**
   - Text: "The tool shall be able to print a DEA in RTF format.

5. **Import DEA's**
   - Text: "The tool shall provide the ability to batch import DEA's from Excel and comma separated value formatted files.

**Notes:**
- Dea lin g, u pda ting and delet ing users.
- Maintaining User Accounts and Database Management.
- **Change Lifecycle State**
- **DEA Lifecycle**
- **Query DEA Database**
- **Create, Update, Delete DEA's**
- **Electronic Signature**
- **List and Print DEA's**
- **Import DEA's**
Conclusion

- **Why you don’t want to model**
  - Modeling is hard
  - Modeling tools are difficult
  - Modeling will likely require cultural changes

- **Why you do want to model**
  - It increases the rate of communications
  - It increases the precision of communications
  - It reduces tacit information
  - It promotes a common understanding of your project

- **Benefit:** Stakeholders having a common understanding of how the project is organized and its objectives will work together more effectively and make better decisions.
Theoretical aspects of modeling

- **SysML and project management**
- **Modeling in the context of SysML**
- **SysML structure and semantics**
- **Conclusion**
While project management and systems engineering share many of the same qualities, there are important differences.
Similarities between PM & SE

- Lots of **behaviors** having complex relationships
- Lots of **entities** having complex relationships
- Many relationships and dependences between behaviors and entities.
Systems engineering

- Systems engineering is a multidisciplinary approach for developing balanced systems solutions in response to diverse stakeholder needs.

- It includes the application of both management and technical processes to achieve balance and mitigate project risk.

- The management process is applied to ensure that development cost, schedule and technical performance objectives are met.
## Differences between PM & SE

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Project Management</th>
<th>Systems Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduling</td>
<td>Absolute times</td>
<td>Relative times</td>
</tr>
<tr>
<td>Economic cost</td>
<td>Mandatory/aggregation centric</td>
<td>Optional/unit centric</td>
</tr>
<tr>
<td>People</td>
<td>Individuals and roles</td>
<td>Role centric</td>
</tr>
<tr>
<td>Entities and behaviors</td>
<td>Notional</td>
<td>Precise</td>
</tr>
</tbody>
</table>

[www.intrinsyx.com](http://www.intrinsyx.com)
Project Management Models

Temporal Models

Gantt, PERT Charts

Structural Models

Behavioral Models

SysML

www.intrinsyx.com
Being a successful PM or SE means:

- Understanding the project/system information and how it fits together.

- Communicating and otherwise making this information available to stakeholders in a timely, consistent fashion in a form relevant to their backgrounds and needs.
Part II, SysML

- SysML and project management
- Modeling in the context of SysML
- SysML structure and semantics
- Summary
SysML modeling is about communications

- It’s a **language** having syntax and structure…
- that uses a **medium**, primarily graphics…
- and has a **methodology**, which currently is largely undefined.
Three main attributes of languages

- Abstractions of the world around us
- Some form of persistence
- A shared experience: a producer and a consumer
Three main attributes of SysML

- Abstractions of systems
- Database persistence
- A shared experience
Model, model on the wall...

who’s the truest of us all:

- An electrical schematic of a radio
- An economic model
- A model student
- A non working model airplane
- A novel about present day life the author believes to be possible
- A description of a pencil

www.intrinsyx.com
SysML and other modeling constructs

Abstraction

More

Concepts

Enterprise architectures

SysML

Domain specific languages

Examples

DoDAF, MoDAF, UPDM, FEA

SysML

UML, Modelica, Simulink, MARTE

Less

www.intrinsyx.com
Abstraction levels

La Joconde

Femme au Chapeau Orné

www.intrinsyx.com
Two criticisms of SysML...

- **Its old ideas warmed over** – Abstractions and persistence have been around at least since humans drew pictures on cave walls. **There is nothing new here.**

- **Its simply not practical** – Having my team think abstractly in the same way and put their information into a database in the same fashion is absurd. **It is not workable.**
A historical perspective
Part II, SysML

- SysML and project management
- Modeling in the context of SysML
- SysML structure and semantics
- Summary
SysML Diagram Taxonomy

SysML Diagram

Behavior Diagram
- Activity Diagram
- Sequence Diagram
- State Machine Diagram
- Use Case Diagram

Requirement Diagram

Structure Diagram
- Internal Block Diagram
- Package Diagram

- Parametric Diagram

Same as UML 2
Modified from UML 2
New diagram type

Source: OMG Specification
Model elements

- Models consist of elements
- Elements must have unique names, within a namespace
- All model elements must reside in (be owned by) one package
Major types of model elements

- Structural
- Behavioral
Packages

- **Models are composed** of one or more packages
- **Packages can contain** other packages and/or any collection of model elements
- **Package hierarchies** define the name space
Packages shown as a containment tree

Browser view of the model’s package hierarchy.
FIGURE 5.1
An example package diagram.
Dependences and stereotypes

**FIGURE 5.10**
Example of dependencies in the camera performance view.
Blocks

- Blocks represent structural elements
  - Organizations
  - Data
  - People
  - Airplanes

- Blocks have two main attributes
  - Properties, other structural elements
  - Behavior, either as an intrinsic capability or through behavioral elements
Block property types

- Parts (in UML, composition)
- References (in UML, aggregation)
- Associations
The black diamond is used for communicating a Part Property relationship.

Figure B.15 - Defining the Automotive Domain (compare with Figure B.4) - (Block Definition Diagram)
No adornment is used for communicating an Association Property relationship.

Figure B.16 - Defining Structure of the Hybrid SUV System (Block Definition Diagram)

Source: OMG Specification
The white diamond is used for communicating a Reference Property relationship.

A reference association on a block definition diagram.
An automobile with four wheels described as separate parts.
Blocks and inheritance

Example of block specialization.

FIGURE 6.35
Activity Diagrams

Define Recipient

Join Decision

Think of Present

[Book or Music]

Buy Online

Get Money

Buy from Store

Merge Fork

[Other]

Wrap Present

Present

Write Gift Card

Join
Sample activity diagram 2

An example activity diagram.
The entire activity is conducted within the JRA workflow system.

If a technical review is not required, then the disposition step function is performed as part of this step.

Email notifications shown here are the minimum. In fact, users will generally be emailed each time the issue is updated or its status changes.
Requirement diagram—Purpose

- Fully and unambiguously specify what the model is to do (functional requirements) and the context in which it is to operate (non-functional requirements).
- Provide concise and unambiguous information showing how requirements relate to each other.
- Provide concise and unambiguous information showing how requirements relate to other model elements—the project lifecycle.
The vehicle shall accelerate from 0 to 60 mph in...
Requirements dependences

- «refine»
- «satisfy»
- «deriveReqt»
- «copy»
- «verify»
- «trace»
Refine, satisfy & deriveReqt

A master cylinder shall have a reservoir compartment for each service brake subsystem serviced by the master cylinder. Loss of fluid from one compartment shall not result in a complete loss of brake fluid from another compartment. ID = "S5.4.1"

The best-practice solution consists in assigning one reservoir per brake line.

The best-practice solution consists in using a set of springs and pistons to confine the loss to a single compartment.

Figure 16.4 - Links between requirements and design
Copy

Figure 16.6 - Use of the copy dependency to facilitate reuse
Verify

req [Package] Testing [verification example]

«requirement»
All Weather Operation

verifiedBy
«interaction» Water Spray Test

«testCase»
Water Spray Test

verdict: VerdictKind
...

verifies
«requirement» All Weather Operation
id = "S1"
text = "The system shall be capable of detecting intruders 24 hours per day, 7 days per week, under all weather conditions."
Use case scope

Use Case Scope

High Level Business Objectives

- Developers describe general system capabilities
- Management articulates overall business objectives

- Developers imagine what/how users will want to do to get their work done.
- Users specify what they need to do to get their work done that day

- Development managers describe to developers how the work will be implemented
- Users describe to developers how the work is to be implemented

Implementation Details

www.intrinsyx.com
Use cases and actors

- Actors are **external entities** that interact with your system via use cases.

- Actors can be **people, computer systems** or really any **device** that interacts with your system.

- It is high recommended that Actors be modeled before use cases.
Actors for a household model

- Adult
- Child
Use cases for household food processes

- Adult
  - Optional participation
  - Household Member
- Child
  - Household Member

- Food Processes
  - Buy Food
  - Prepare Meal
    - Prepare Breakfast or Lunch
  - Eat Meal
    - Include
  - Prepare Dinner
    - Extend
  - Formal Dinner
  - BBQ Dinner
  - Clean Up After Meal
Part II, SysML

- SysML and project management
- Modeling in the context of SysML
- SysML structure and semantics
- Summary
Key SysML features & capabilities

- Open standard
- Works equally well with structural and behavioral artifacts
- Treats requirements as first-class modeling elements
- Future plans include simulations
- Formal mechanism for extending SysML semantics
- Formal set of semantically consistent graphical elements.

www.intrinsyx.com
SysML requirements relationships

Any behavioral diagram.

SysML Requirements Dependency Relationships

- <<refine>>
- <<verify>>
- <<allocate>>
- <<satisfy>>
- <<trace>>
- <<allocate>>
- <<block>>
- <<refine>>

Any entity executing the activity: person, org, machine, software, etc.

Id = ""
Text = ""

www.intrinsyx.com
Conclusion

- **Why you don’t want to model**
  - Modeling is hard
  - Modeling tools are difficult
  - Modeling will likely require cultural changes

- **Why you do want to model**
  - It increases the rate of communications
  - It increases the precision of communications
  - It reduces tacit information
  - It promotes a common understanding of your project

- **Benefit:** Stakeholders having a common understanding of how the project is organized and its objectives will work together more effectively and make better decisions.

www.intrinsyx.com
An ad hoc initial attempt at modeling the NASA Constellation program, manned moon/Mars mission.

The notions and ideas expressed in these slides represent my own interpretations and have not been vetted or sanctioned by NASA in any way. They are presented solely for educational purposes regarding how large engineering projects might be rendered in a SysML model.
## Constellation Program Model

### Table of Contents

#### Structural Artifacts

<table>
<thead>
<tr>
<th>Organization Information</th>
<th>Data Taxonomy</th>
<th>Information Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>NASA Organization</td>
<td>High Level Structure</td>
<td>Tools</td>
</tr>
<tr>
<td>Constellation Organization</td>
<td>Operational Data</td>
<td>Tool Org and Functions</td>
</tr>
<tr>
<td>SE&amp;I</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP&amp;C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Behavioral Artifacts

### Virtual Missions and Operations

- GMIP Category Temporal Decomposition
- GMIP Category Information Decomposition
- Design Category Decomposition
- EVA H/W Processing & Crew Training
- Integrated Vehicle Engineering Analysis
- MPPF
- Offline Processing
- CMISM Stack Integration
- VAB Integrated Operations

www.intrinsyx.com
Generic Mission Template
Integrated Vehicle Engineering Analysis

Duration for these concurrent activities is L-210 days to L-120 days
Constellation Data Taxonomy

Root

Document
- Requirement Document
  - dataType = "document"
- Procedure
  - dataType = "document"

Model

Process Record
- Data Exchange Agreement
  - dataType = "processRecord"
- Change Request
  - dataType = "processRecord"
Relationship between UML and SysML

- UML reused by SysML (UML4SysML)
- SysML extensions to UML (SysML Profile)
- UML not required by SysML (UML - UML4SysML)

Source: OMG Specification
Figure 7.8 - An example of the four-layer metamodel hierarchy
Copyright Information

OMG Source material is reprinted with permission. Object Management Group, Inc. (C) OMG. 1997 - 2006.