Moving from Documents to Models

Dr. Darius Silingas, Chief Solutions Architect @ No Magic, Inc.

MBSE webinar
February 23, 2016
Evolution of Information Exchange
State of Practice in Systems Engineering: Documents Rule!

Document based approach works, but...

According to research, statistically 55% highest level severity defects are made in Requirements Analysis and Design phases.
MBSE is about driving systems engineering by modeling & models
Document-Based SE versus MBSE

SE Practices for Describing Systems

**Past**
- Specifications
- Interface requirements
- System design
- Analysis & Trade-off
- Test plans

**Future**

Moving from Document-centric to Model-centric
Model vs. Drawing

Model vs. Drawing

Reuse

Model Repository

Generated Document

Hard Copies

Document with Drawing
Getting Value from Models and Modeling

- Manage complexity
- Improve quality
- Improve communication
- Enable closer cross-functional collaboration
- Preserve knowledge
- Reuse
Modeling Solution is a combination of a modeling language(s), a methodology and a modeling tool that together provide a productive infrastructure for applying model-driven development in the context of a particular organization.
OMG Systems Modeling Language (SysML) is a graphical modeling language for specification, analysis, design, verification and validation of systems.

- Modeling language that includes 9 diagrams.
- Dedicated for modeling complex systems that may include hardware, software, information, personnel, procedures, facilities, etc.
The Four Pillars of SysML
The modeling language is just the language, and must be combined with a methodology to be useful.
System Context and Requirements

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Text</th>
<th>Verify Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Environment Temperature</td>
<td>The system should function within environment temperature range between -10°C and +50°C.</td>
<td>Test</td>
</tr>
<tr>
<td>2</td>
<td>Coffee Temperature</td>
<td>The system should produce coffee, which temperature is 80 +/- 5°C.</td>
<td>Test</td>
</tr>
<tr>
<td>3</td>
<td>Robust Corpus</td>
<td>The system should be resistant to vandalism actions.</td>
<td>Test</td>
</tr>
<tr>
<td>4</td>
<td>Make Coffee</td>
<td>The system shall produce coffee.</td>
<td>Demonstration</td>
</tr>
<tr>
<td>5</td>
<td>Coffee Types</td>
<td>The system shall provide choice of coffee kinds.</td>
<td>Demonstration</td>
</tr>
<tr>
<td>6</td>
<td>Cup Size</td>
<td>The system shall provide ability to user to choose cup size (small, medium, large).</td>
<td>Demonstration</td>
</tr>
<tr>
<td>7</td>
<td>Locked Corpus</td>
<td>The system shall not allow unauthorized servicing.</td>
<td>Demonstration</td>
</tr>
<tr>
<td>8</td>
<td>Coffee in 10s</td>
<td>The system shall be able to produce coffee in 10s.</td>
<td>Test</td>
</tr>
<tr>
<td>9</td>
<td>Size Limits</td>
<td>The system size should not exceed 200x100x50.</td>
<td>Inspection</td>
</tr>
<tr>
<td>10</td>
<td>Ease of Maintenance</td>
<td>The system shall be easy to maintain.</td>
<td>Analysis</td>
</tr>
<tr>
<td>11</td>
<td>Pay with Credit Card</td>
<td>The system shall allow to pay for coffee by credit card.</td>
<td>Demonstration</td>
</tr>
</tbody>
</table>
Use Case Analysis
System Specification (Black Box)

[Diagram showing System Model with States and Transitions]

- **System** CoffeeMachine
  - **CoffeeMachine**
    - **Flow Properties**
      - **in request**: CoffeeRequest
      - **out options**: CoffeeOptions
      - **in choice**: CoffeeChoice
      - **suppliesLevel**: Integer
      - **servicingSuppliesLevel**: Integer = 10
      - **needsServicing**: Boolean
  - **Interface Blocks**
    - **iCoffeeMachinePOS**
      - **flow properties**
        - **in request**: CoffeeRequest
        - **out options**: CoffeeOptions
        - **in choice**: CoffeeChoice
    - **iPaymentHandler**
      - **flow properties**
        - **in transaction**: Transaction
        - **out result**: TransactionResult
    - **iCoffeeMachinePayment**
      - **flow properties**
        - **out request**: PaymentRequest
        - **in payment**: PaymentInfo

- **State Machine** CoffeeMachine Lifecycle
  - **States**
    - **Ready**
      - **entry / Initialize Coffee Machine**
      - **after (30s)**
        - **CoffeeRequest**
      - **Selling**
        - **do / Sell Coffee**
        - **PaymentReceived**
      - **Brewing**
        - **do / Brew Coffee**
        - **exit / suppliesLevel--**
      - **Dispensing**
        - **do / Dispense Coffee Cup**
  - **Transitions**
    - **when (suppliesLevel < 1)**
      - **Empty**
        - **entry / Request Supplies Refill**

- **Par** [System] CoffeeMachine
  - **suppliesLevel**: Integer
    - **servicingSuppliesLevel**: Integer
    - **needsServicing**: Boolean
    - **constraint**: BelowThreshold
      - **{below = value < threshold}**

[Diagram details and relationships between states and transitions are omitted for clarity.]
System Design (White Box)
### Establishing Traces between Elements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Brew Coffee</th>
<th>Dispense Coffee Cup</th>
<th>Initialize Coffee Machine</th>
<th>Request Supplies Refill</th>
<th>Sell Coffee</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Coffee Temperature</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 Make Coffee</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Coffee Types</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 Cup Size</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 Pay with Credit Card</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Inform about Limited Supplies</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 Inform about Need to Remove Waste</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Coffee Brewer
- Brew Coffee
- Dispense Coffee Cup
- Initialize Coffee Machine
- Request Supplies Refill
- Sell Coffee

### Coffee Cup Dispenser
- Coffee Brewer

### Coffee Machine Payment
- Coffee Cup Dispenser

### Coffee Machine POS
- Coffee Machine Payment
Need for a Method/Framework

- This opens discussions of:
  - how to structure the model
  - what views to build
  - which artifacts to deliver
  - and in what sequence

Every company deals with the same issues differently. Some use:
- defense architecture frameworks: DoDAF, NAF, MODAF
- MBSE methods: OOSEM, Harmony, SYSMOD, FAS; however, saying there is no need for an architectural framework just doesn’t work.
# Magic Grid

<table>
<thead>
<tr>
<th>Layer of Abstraction</th>
<th>Pillar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Requirements</td>
</tr>
<tr>
<td>Concept</td>
<td>C1</td>
</tr>
<tr>
<td>Stakeholder Needs</td>
<td>Use Cases</td>
</tr>
<tr>
<td>Problem</td>
<td>P1</td>
</tr>
<tr>
<td>Goals &amp; Objectives</td>
<td>Functional Analysis</td>
</tr>
<tr>
<td>Solution</td>
<td>S1</td>
</tr>
<tr>
<td>Component Requirements</td>
<td>Component Behavior</td>
</tr>
</tbody>
</table>
You always end up using an architecture framework whether you want or don’t
Systems Modeling Platform

- A modeling platform for system architects, system analysts, developers, quality engineers, etc.
- One of the most popular UML / SysML tools in the market
- Available since 1998
- Over 1,000,000 installations in 90+ countries
- Widely regarded as the most standard-compliant tool
- Designed for customization to customer needs

More info: www.magicdraw.com

Awards
Collaborative Modeling

Collaborate in teams using MagicDraw Teamwork Server!

- Centralized repository
- Controlled access to models
- Multiple users working simultaneously on the same model
- Prevention of conflicts
- Configuration management (versions, branching, comparison, merge)
Realizing the Value of Modeling and Models

Creating Models

Center of Excellence

Principles

Governing Models

Using Models

High Value from Modeling and Models
Making Architecture Available to Users

Models must be accessible without specialized tools!

- Multiple points of view
- Multiple forms
  - HTML for browsing
  - Word for approving

http://webreports.nomagic.com
Publish Architecture for Browsing in Web

http://webreports.nomagic.com
Cameo Collaborator

Collaborate in Teams by Reviewing Published Models

Author via MagicDraw

Publish project

Share published project with reviewers

React to reviewer’s feedback

Reviewer via Cameo Collaborator

Review project and provide feedback
Transition is NOT Easy
Adopting MBSE

- Gain Understanding
- Build Proof of Concept
- Run Pilot Project
- Institutionalize a New Practice
Top 10 Lessons Learned

1. Without Leadership, Organizations Don’t Change
2. Establish Internal Center of Competence
3. Dedicate Time for Coaching and Learning
4. SysML is Not Enough, Methodology is a Critical Factor for Success
5. Start with a Small Team, and Achieve Quick Wins
6. Clearly Choose which Problems You Want to Solve with MBSE
7. Enable Model-based Cross-Functional Collaboration
8. Investigate how to Exchange Data Between Modeling and Other Tools
9. Choose Only One Tool as a Master Source for Certain Information
10. Communicate, Communicate, Communicate ...
The Recipe for Success

think **BIG**

start **SMALL**

and **EVOLVE**
Let’s Keep in Touch

Dr. Darius Silingas
Chief Solutions Architect
No Magic Europe
Phone (direct): +370 37 705899
e-mail: darius.silingas@nomagic.com
lt.linkedin.com/in/dariussilingas/