SE Use Cases
SysML - SE DSIG

John Watson
Lockheed Martin
2/24/2014
Sources for Use Cases

• Used the SEBoK to identify
  – Description of Knowledge Areas
  – Organized based on 15288 Life Cycle Stages

• ISO/IEC 15288
  – Definition of Life Cycle Stages
  – Content of Life Cycle Stages
Organize Use Cases by Life Cycle Phases

- Exploratory/Concept Stage
- System Development Stage
  - Management Use Cases
  - SE Domain Use Cases
  - Validation and Verification Use Cases
- Production Stage
- Product and Service Life Management Stage
System Architecture Domain Activities

- Analyze Stakeholder Needs
- Analyze Missions
- Analyze System Behavior
- Derive system requirements
- Derive logical and physical structure
- Derive System Components Specifications
- Manage System Life Cycle Costs
SE Domain Integration and Information Layers

- Each SE Domain;
  - Contributes domain specific information
  - Has Responsibility for their information content
  - Information content can reference content from other domains
  - Iterates solution with other Domains
  - Has one or more views to information content
  - Defines and manages Requirements
  - Measures impact of their changes in their domain and across other domains
  - Conducts Reviews
  - Produces Deliverables

System Architecture Definition Domain

Mission Analysis Domain
Infrastructure Engineering Domain
SWaP Management Domain
RMA Management Domain
Structural Analysis Domain
Security Engineering Domain
Safety Engineering Domain
Performance Analysis Domain
Verification and Validation Domain
Human Systems Integration Domain
Environmental Engineering Domain
Development Management Domain
To measure SysML effectiveness we need to understand the context of how it is used.
Suggested Approach

• Define the System Engineering Development Context
  – Not as set of independent tools

• Identify the SE Use Cases this System must support
  – First Pass - Identifying Use Case Goal, Primary Actor and Textual Description
  – Select the few that will provide the most SysML benefit
    • Derive Functional entities, via Activity Diagrams
    • Derive SysML Requirements
    • Determine to what extent is SysML supporting System Engineering
Future Use

• Architect a Development System
  – Decompose System into a set of Components with Interfaces
  – Components and Interfaces are standards based, e.g. OSLC

• Provide input, clarity and vision to:
  – Standards Groups –
    • Identifying enhancements to existing standards
    • Demonstrate the need of integration threads across standards
    • Identify new standards
  – Tool Vendors –
    • A full view of the need
    • A specification of what needs to be built Tool Vendors
Life Cycle System Engineering Use Cases

• **Exploratory/Concept Stage**
  – Evaluate Customer Proposal
  – Define Stakeholders Needs
  – Analyze System Missions
  – Define the System Requirements
  – Analyze System Life-cycle Costs

• **System Development Stage**
  – **Management Use Cases**
    • Plan a Development Cycle
    • Manage Development Progress
    • Manage Development Environment
Life Cycle System Engineering Use Cases

• System Development Stage (Continued)
  – SE Domain Use Cases
    • Derive Product Architecture
    • Evaluate System Safety
    • Perform System Reliability, Availability and Maintainability Engineering
    • Perform System Security Engineering
    • Analyze System Performance
    • Allocate and Manage SWaP
    • Perform Trade Study
    • Analyze Behavior Correctness
    • Manage Product Lines
    • Integrate Human Domain Constraints
    • Perform Environmental Engineering
    • Integrate with Implementation Domains
    • Perform EMI Engineering
• System Development Stage (Continued)
  – Validation and Verification Use Cases
    • Develop Verification Plan and Procedures
    • Develop a System Integration Plan
    • Execute a Verification Test Procedure
    • Provide V&V Status

• Production Stage Use Cases
  – Support Produce-ability Engineering

• Product and Service Life Management
  – Support Initial Installation
  – Evaluate Change Request
  – Support System Modernization Plan
  – Support System Disposal and Retirement
Activity - Perform System Security Engineering

- **Goal** – The goal of this use case is to incorporate in the system of interest the necessary security design features to meet the needs of the customer.
- **Primary Actor** – SE Security Specialist
- **Secondary Actors** –
- **Preconditions** –
  1. A list of known potential threats are available
  2. A list of applicable policy documentation is available
- **Activity** – This use case begins early in the development cycle and continues to iterate through the remaining development cycles as the product matures.
  1. Obtain and/or define the customer’s security protection goals for the following security domains including:
     1. Information security governance and risk management
     2. Access control
     3. Cryptography
     4. Physical (environmental) security
     5. Security architecture and design
     6. Business continuity and disaster recovery planning
     7. Telecommunications and network security
     8. Application development security
     9. Operations security
     10. Legal, regulations, investigations, and compliance
  2. Capture the system vulnerabilities by analyzing the known or perceived threats and their behavior.
3. Derive a set of security requirements that address the vulnerabilities and other applicable security policy documents.

4. Evaluate points of Interface;
   1. Identify all external interface points
   2. Identify internal interface points of major subsystems such as server farms, sensors, security management, business network, etc.
   3. Identifying the points of interface may have been completed earlier in a use case such as “Derive Product Architecture”.
   4. Determine and capture the level of security required for the information exchanged at the points of interface.

5. Capture the security architecture design that satisfy these requirements and minimize or contain the vulnerabilities.

6. Measure the change impact to other domains and mitigate issues

7. Conduct appropriate reviews within engineering and with the customer

8. Capture test cases that validate the security requirements have been reached.

9. If the proposed design does not meet the System goals, refine the design.

10. Prepare the necessary documentation for system accreditation and certification.

Post Conditions – Accreditation Certificate is submitted
Perform System Security Engineering UC

Create and share validated reference libraries

View external reference documents

1. Define and Organize UCs
2. Create Activities to expose vulnerabilities
3. Integration with threat analysis tools

1. Define and Organize Requirements
2. Create Activities

Update Structure and interface definitions

Verify requirements are satisfied with test cases

1. Define a document structure and content
2. Publish document

Create a domain specific profile that includes concepts and iconic representations compatible with SysML

1. Define and Organize Requirements
2. Create Activities

1. Assessing change impacts
2. Include and overlay impacted domain views
3. Remove un-impacted domain views

Execute defined behavior
Artifact Review Pattern

1. Create a change request
2. Include Reviewer and Date
3. Submit changes to Reviewee

Create a view to isolate information to be reviewed
Share information to be reviewed
Manage and track change requests
Update and share reviewed information view
Manage and merge multiple branches
Baseline model artifacts
Summary

• Examine the complete System Engineering Context to examine:
  – How well is SysML supporting System Engineering activities?
  – Are there other areas where SysML could be expanded?

• Use SysML to:
  – Define that Context
  – Define System Engineering Use Cases
  – Drive the language requirements

• One Use Case Example was shown but;
  – We expect to see re-occurring patterns and requirements throughout many of the use cases
References


What to Harvest from Use Cases

• Functional entities
  – Represents the Development System functionality required by Systems Engineers to do their work
  – Examine how SysML is used to support each functional entity
  – Many will appear in multiple Use Cases
  – Use these functional entities to derive SysML Requirements

• Functional Entity Examples;
  – Conduct a Review, Capture System behavior/structure/requirements, measure change impact, share information across domains, produce a deliverable, analyze performance, select a domain view, select a multi-domain view, create a baseline, assess change impact, manage domain information, etc.
Infrastructure Engineering Domain Activities

- Define Hardware platforms and performance
- Define Physical Network
- Define System Management
  - Define Status and Error Messages
  - Status and error collection and reporting
  - Error management
- Define common system services
- Time management
- Redundancy Architecture
Industry Available Product Phases


Typical High-Tech Commercial Systems Integrator

Typical High-Tech Commercial Manufacturer

US Department of Defense (DoD) 5000.2

NASA

US Department of Energy (DoE)