System Engineering Workflow Use Cases

Use Case Survey

Last Updated: Wednesday, September 7, 2014 4:09:08 PM

Table of Contents

[1. Introduction 5](#_Toc397869731)

[1.1. Background 5](#_Toc397869732)

[1.2. Approach for Creating SE Workflow Use Cases 6](#_Toc397869733)

[1.3. Assumptions 7](#_Toc397869734)

[2. Terminology 8](#_Toc397869735)

[3. Context 10](#_Toc397869736)

[4. Life Cycle Workflow Use Cases for System Engineering 11](#_Toc397869737)

[4.1. Exploratory and Concept Stage Workflow Use Cases 12](#_Toc397869738)

[**4.1.1.** **Plan a Response to a Customer Request** 12](#_Toc397869739)

[**4.1.2.** **Analyze Stakeholders Needs** 13](#_Toc397869740)

[**4.1.3.** **Derive System Requirements** 14](#_Toc397869741)

[**4.1.4.** **Analyze System Life-cycle Costs** 15](#_Toc397869742)

[**4.1.5.** **Workflow Use Case Activity Diagrams** 17](#_Toc397869743)

[***4.1.5.1.*** ***Plan a Response to a Customer Request*** 17](#_Toc397869744)

[***4.1.5.2.*** ***Analyze Stakeholders Needs*** 19](#_Toc397869745)

[4.2. System Development Stage Workflow Use Cases 20](#_Toc397869746)

[**4.2.1.** **Management Workflow Use Cases** 20](#_Toc397869747)

[***4.2.1.1.*** ***Plan a Development Cycle*** 20](#_Toc397869748)

[***4.2.1.2.*** ***Manage Development Progress*** 21](#_Toc397869749)

[***4.2.1.3.*** ***Manage Development Environment*** 21](#_Toc397869750)

[***4.2.1.4.*** ***Create a Baseline*** 23](#_Toc397869751)

[**4.2.2.** **SE Domain Workflow Use Cases** 24](#_Toc397869752)

[***4.2.2.1.*** ***Derive Product Architecture*** 26](#_Toc397869753)

[***4.2.2.2.*** ***Evaluate System Safety*** 28](#_Toc397869754)

[***4.2.2.3.*** ***Perform System RMA Engineering*** 28](#_Toc397869755)

[***4.2.2.4.*** ***Perform System Security Engineering*** 29](#_Toc397869756)

[***4.2.2.5.*** ***Analyze System Performance*** 30](#_Toc397869757)

[***4.2.2.6.*** ***Allocate and Manage SWaP*** 30](#_Toc397869758)

[***4.2.2.7.*** ***Perform a Trade Study*** 31](#_Toc397869759)

[***4.2.2.8.*** ***Analyze Behavior Correctness*** 31](#_Toc397869760)

[***4.2.2.9.*** ***Manage Product Lines*** 31](#_Toc397869761)

[***4.2.2.10.*** ***Integrate Human Domain Constraints*** 32](#_Toc397869762)

[***4.2.2.11.*** ***Perform Environmental Engineering*** 32](#_Toc397869763)

[***4.2.2.12.*** ***Collaborate with Implementation Domain Team*** 32](#_Toc397869764)

[***4.2.2.13.*** ***Preform EMI Engineering*** 32](#_Toc397869765)

[***4.2.2.14.*** ***SE Domain Workflow Use Case Activity Diagrams*** 34](#_Toc397869766)

[**4.2.3.** **Validation and Verification Workflow Use Cases** 36](#_Toc397869767)

[***4.2.3.1.*** ***Develop Verification Plan and Procedures*** 36](#_Toc397869768)

[***4.2.3.2.*** ***Develop a System Integration Plan*** 37](#_Toc397869769)

[***4.2.3.3.*** ***Execute a Verification Test Procedure*** 37](#_Toc397869770)

[***4.2.3.4.*** ***Provide V&V Status*** 37](#_Toc397869771)

[4.3. Production Stage Workflow Use Cases 38](#_Toc397869772)

[**4.3.1.** **Support Produceability Engineering** 38](#_Toc397869773)

[4.4. Product and Service Life Management Workflow Use Cases 38](#_Toc397869774)

[**4.4.1.** **Support Initial Installation** 39](#_Toc397869775)

[**4.4.2.** **Architect Sustainability System** 40](#_Toc397869776)

[**4.4.3.** **Evaluate Change Request** 40](#_Toc397869777)

[**4.4.4.** **Support System Modernization Plan** 40](#_Toc397869778)

[**4.4.5.** **Support System Disposal and Retirement** 41](#_Toc397869779)

[4.5. Supporting Activity Behaviors 42](#_Toc397869780)

[**4.5.1.** **Measure a Change Impact** 42](#_Toc397869781)

[**4.5.2.** **Conduct a Review** 44](#_Toc397869782)

[**4.5.3.** **Import Reference Materiel into Model** 45](#_Toc397869783)

[**4.5.4.** **Perform a Trade Study** 46](#_Toc397869784)

[5. References 46](#_Toc397869785)

[6. Addendum 47](#_Toc397869786)

[6.1. Workflow Use Case Templates 47](#_Toc397869787)

[**6.1.1.** **Workflow Use Case Pattern Template 1** 47](#_Toc397869788)

[**6.1.2.** **Workflow Use Case Pattern Template 2** 47](#_Toc397869789)

[6.2. Industry Available Product Phases 48](#_Toc397869790)

Table of Figures

[**Figure 1: Terminology Relationships** 8](#_Toc397869791)

[**Figure 2: Systems Engineering Workflow Use Case Context** 10](#_Toc397869792)

[**Figure 3: Use Case Actors** 11](#_Toc397869793)

[**Figure 4: Exploratory and Concept Stage Workflow Use Cases** 12](#_Toc397869794)

[**Figure 5: Plan a Response to a Customer Request** 18](#_Toc397869795)

[**Figure 6: Analyze Stakeholders Needs** 19](#_Toc397869796)

[**Figure 7: Management Workflow Use Cases** 20](#_Toc397869797)

[**Figure 8: SE Domain Workflow Use Cases** 24](#_Toc397869798)

[**Figure 9: Specialty Engineering** 26](#_Toc397869799)

[**Figure 10: Analysis Workflow Use Cases** 26](#_Toc397869800)

[**Figure 11: Security Engineering Task** 35](#_Toc397869801)

[**Figure 12: Validation and Verification Workflow Use Cases** 36](#_Toc397869802)

[**Figure 13: Production Stage Workflow Use Cases** 38](#_Toc397869803)

[**Figure 14: Product and Service Life Management Stage Workflow Use Cases** 39](#_Toc397869804)

[**Figure 15: Measure a Change Impact** 43](#_Toc397869805)

[**Figure 16: Conduct a Review** 45](#_Toc397869806)

[**Figure 17: Import Reference Materiel into Model** 45](#_Toc397869807)

[**Figure 18: Perform a Trade Study** 46](#_Toc397869808)

[**Figure 19: Industry Available Product Phases** 48](#_Toc397869809)

1. **Introduction**
	1. **Background**

The motivation to create this document surfaced during the planning of the OMG SysML 1.5 RTF activity. This planning activity determined a need to assess the degree to which SysML has addressed the original "UML for SE RFP" requirements and to define a roadmap for the continuing evolution of the SysML language. One of the approaches for this assessment was to develop a set of Systems Engineering (SE) Workflow Use Cases.

The intent of the SE Workflow Use Cases is to capture the needed capabilities and infrastructure required by SE to develop large complex systems across the entire product lifecycle. Therefore the context of these use cases will be the Systems Engineering Development System (SEDS). The SEDS's use cases, the system context, system domain and system structure will be captured within a SysML model using some of the workflows described in the SE workflow use cases. The longer-term intent of this model is to derive a reference architecture of a fully integrated SEDS, where each derived component incorporates important design considerations such as separation of concerns, encapsulation, maximum cohesion, minimal coupling and a defined interface. The SEDS architecture should include all aspects of system design considerations including reliability, security, maintainability, usability, extensibility, availability, extensibility, etc.

This SEDS reference architecture then can be used by our tool vendors to help identify products that are needed to support and automate this MBSE development environment. In addition, this information can be used by industry standards bodies to identify what industry standards are needed and why.

Of course the immediate need is to help define the roadmap for future SysML enhancements. The SE use cases should justify existing features of SysML and help discover additional needs including areas such as model presentation; model construction and change; interoperability with other models and other tools;, managing the model content and managing the model content through change; analyzing the model content within a model or analyzing this content with other models and tools.

Without this SEDS context understanding the needs to perform SE would be very difficult to develop a SEDS environment that would be fully integrated, cost effective, automated, and highly productive. Today we struggle trying to integrate many disparate tools. We often fall on point-to-point tool integration solutions that add complexity and maintenance issues that distract us from our intended workflow and are costly to create and maintain.

SysML was developed to provide a common SE language that supports the specification, analysis, design, verification, and validation of a broad range of systems, especially complex systems. Initially the primary use of SysML within Systems Engineering was to derive and capture a system's architecture and flowdown of requirements within this context. These architectural models provided the traceability from the original customer needs to all structural and behavioral aspects of the system and therefore rationalize all aspects of the system's behavior and structure back to the original customer needs.

As the use of SysML progressed it became more apparent that the SysML Architectural System model could provide even more value. Some of the earlier observations saw value with integrating the implementation disciplines, including the software, mechanical and electrical engineering disciplines. Specialty engineering implementation disciplines could also be layered on the foundational system architectural model and extend linkages to other specialty tools and artifacts. These specialty-engineering aspects include security, safety, integration and test, SWaP (Size, Weight and Power), RMA (reliability, maintainability and accessibility) network management, environmental engineering, performance analysis and more.

These informational threads allow us to:

* Easily share, communicate and exchange information with all engineering disciplines
* Effectively measure change impact across these disciplines and throughout the entire life cycle of the product
* Minimize redundant information by defining information once within the most appropriate domain and sharing this information across all domains that required it
* Create an environment for automation and validation of the information content

With such an environment the architectural model becomes a significant engineering information hub providing a path to connect all engineering information back to the customer needs.

* 1. **Approach for Creating SE Workflow Use Cases**
1. A SysML Roadmap Workflow Use Case Team will be brought together. This team will be responsible for managing the capture of a set of SE Workflow Use Cases within a SysML model.
2. The initial model effort will:
	1. Define a context for a set of SE workflow use cases. This context will be called the Systems Engineering Development System (SEDS).
	2. Capture a set of SE workflow use cases. This set of use cases will be referred to as a Use Case Survey.
	3. The information source used to derive this SE Case Survey will be derived from at least the ISO/IEC 15288-2008, INCOSE Systems Engineering Handbook, and the on-line System Engineering Book of Knowledge (SEBok). Other industry references could be used.
	4. The workflow use cases will span across an entire product life-cycle
	5. As a first step for each use case the team will capture:
		1. The primary actor and the use case of the primary actor and the primary actor
		2. The primary actors can include Systems Engineers, Architects, implementers, analysts, engineering management, test engineers, Program Management, subcontractors, etc.
		3. Additional preliminary information can also be collected at this point if available including preconditions, post conditions, and set of workflow steps needed to achieve the use case goal. This can be represented in a textual form and/or an activity diagram.
	6. The intent of the SE workflow use cases will be to identify "what" steps are taken to achieve each use case goal. At this point, the use case will not describe how each step is realized or. Each use case step could conceivably be realized by a human, by a tool or a combination of both.
3. As the initial use case survey efforts conclude, the SysML Roadmap Use Case Team will solicit help from experts across the industry to help refine and complete each of the use cases.
4. Because of the anticipated large effort involved in creating and maturing these use cases, the use cases will be prioritized.
5. SysML Roadmap Workflow Use Case Team can add, delete or replace use cases at any time during their development.
6. Periodically the progress of this effort will be provided to the full SysML Roadmap Team and to the SysML RTF team.
7. The ultimate intent of this information is to help define the capabilities and infrastructure required by SE to utilize and integrated Systems Engineering across the lifecycle of a products. The benefits include:
	1. This information can be used by our tool vendors to make products that are needed to support an MBSE environment
	2. This information can be used by the industry standards bodies to identify what industry standards are needed and why. The immediate need is to help define the roadmap for future
	3. Without this understanding up front, it would be very difficult to develop a SEDS environment that is highly integrated, cost effective, highly automated, and highly productive.
	4. **Assumptions**
8. The workflow use cases defined are intended to be used on large complex systems supported by large geographically diverse development teams. With smaller and simpler systems some of the use cases may not be needed or, they may require a simpler workflow.
9. The workflow use cases are described assuming a model-based approach is used to develop a system. However, many of the use cases are not dependent on using model-based techniques, since they are the very same use cases System Engineers have been using before model-based techniques were available.
10. **Terminology**

The following table provides a list of terminology used.



**Figure 1: Terminology Relationships**

**Table 1: Terminology Table**

| **Term** | **Acronym** | **Definition** |
| --- | --- | --- |
| Model-based Systems Engineering | MBSE | "Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases".Ref - International Council on Systems Engineering (INCOSE), Systems Engineering Vision 2020, Version 2.03, TP-2004-004-02, September 2007.  |
| Model-based Systems Development | MBSD | Model-based Systems Development (MBSD) is the formalized application of modeling to support all aspects of product engineering and support system requirements, design, implementation, analysis, verification, validation, manufacturing, support and management activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases. Therefore MBSD includes domains such as MBSE, software design and implementation and mechanical design and implementation, and electrical design and implementation.  |
| Product Development System | PDS | Product Development System (PDS) is the system used to provide an integrated environment of tools and capabilities required to develop products that are systems. This includes the environment for systems engineering, software design and implementation engineering, mechanical design and implementation engineering, electrical design and implementation engineering and interfaces to external domains including manufacturing support and product management. |
| Support |  | The organization contracted to provide technical support to the customer for the system-of-interest. This can include installation of the systems, answering technical questions, investigating customer issues, planned maintenance, training and removal at the end of the life cycle.  |
| SE Development System | SEDS | Systems Engineering Development System (SEDS) is the context for the Systems Engineering Workflow Use Cases. It is the system used to provide an integrated environment of tools and capabilities required to perform Systems Engineering activities and tasks. This includes the environment to support system requirements flowdown, design, analysis, verification, validation activities. The SEDS provides interfaces to external domains such as the software, mechanical, electrical engineering domains and interfaces to manufacturing, support and product management domains. |
| Change Management System | CM | A set of functionality that provides capabilities to support a change management process such as managing change requests (opening, grouping, assigning responsibility, managing state, archiving), managing versions of files, defining and managing branches and defining and managing baselines.  |
| Computational Tool | CT | A tool used to provide some type of defined computations for a particular type of analysis. This computational tool could be implemented with a common spreadsheet or a very specialized tool.  |

1. **Context**

The diagram below defines a typical Product Domain structure. Within this domain is the SE Development System. Systems Engineering Development System (SEDS) is the context for the Systems Engineering Workflow Use Cases. It is the system used to provide an integrated environment of tools and capabilities required to perform Systems Engineering activities and tasks. This includes the environment to support system requirements flowdown, design, analysis, verification, validation activities. The SEDS provides interfaces to external domains such as the software, mechanical, electrical engineering domains and interfaces to manufacturing, support and product management domains.



**Figure 2: Systems Engineering Workflow Use Case Context**

1. **Life Cycle Workflow Use Cases for System Engineering**

This section captures the set of systems engineer workflow use cases. Therefore it does not include all use cases for all domains. Often System Engineers are the primary actor for these use cases but the use case can be driven from another stakeholder where the systems engineer is a contributor. The workflow use cases within a lifecycle phase may not be executed in any particular order or are they executed one at a time. Typically within a lifecycle phase, multiple process use cases are being executed in parallel and often one is providing more data refinement to another.



**Figure 3: Use Case Actors**

* 1. **Exploratory and Concept Stage Workflow Use Cases**



**Figure 4: Exploratory and Concept Stage Workflow Use Cases**

* + 1. **Plan a Response to a Customer Request**

**Goal -** The goal of this workflow use case is to create and execute a plan for providing a technical response to a customer's request for a proposal to enhancements an existing system or to produce a new system. **Primary Actor -** SE System Architect**Secondary Actors -** Customer, Program Management, Business Management**Preconditions -** Proposal is received**Activity -**See the accompanying Activity Diagram for the steps executed during this use case.In this use case the proposal is evaluated identifying the technical risks and opportunities. As a result a set of technical tasks are defined, planned and managed that help mitigate the risks and ultimately provide a technical response to the proposal team in the form of a conceptual design. Many of the technical tasks executed during a proposal evaluation phase are similar to those performed during development phases. However, due to time and money constraints, these tasks are limited in scope. Some of these tasks can include: 1. Derive the stakeholder's needs and requirements across the product life cycle 2. Evaluate the customer's missions where the system of interest will play a role 3. Translate the customer's needs to a set of preliminary system requirements 4. Determine the appropriate measures of effectiveness (MOE) and preliminary verification plans 5. Launch necessary research projects to evaluate technology and analyze alternatives 6. Captured and iterated on one or more potential solutions 7. Perform trade studies on candidate ideas and solutions to determine optimal solution 8. Provide an estimate for development costs, development timeline and System costs**Post Conditions -** 1. The required results of the set of technical tasks to respond to this proposal have been captured. 2. The tasks have been planned and executed.

* + 1. **Analyze Stakeholders Needs**

**Goal -** The goal of this workflow use case is to identify all stakeholders and better understand and capture their required needs expectations, goals, and objectives across the entire product life cycle.**Primary Actor -** System Architect**Secondary Actors -** Stakeholders**Preconditions -** **Activity -** 1. Define Stakeholders and their Needs/Goals - In this task the primary actor identifies and captures the system-of-interest stakeholders from across all life cycle stages. The intent is to elicit and capture a set of stakeholder needs, expectations, goals, or objectives for a desired solution. 2. Define the system domain level model - This task is to evaluate and capture the existing domain structure with and without the system-of-interest included. In this task the definition of the appropriate domain entities and their relationship to other entities are captured. These definitions can include stakeholders, other systems, organizations or roles that participate with or influence the system-of-interest. Also captured are the primary interfaces for each. 3. Analyze mission level use cases - Identify the mission level use cases where the system-of-interest will be expected to participate. This task includes capturing the behavior of the system-of-interest and other participating entities to fully understand the needs of the system-of-interest, what constraints will be imposed because of this surrounding environment and the key system-of-interest interfaces. 4. Analyze system threats - Identify and capture any additional use cases or scenarios that may occur during the mission level use cases that can pose an external threat or unwanted system behavior. 7. Define Effective Measures - Elicit from the stakeholders a set of measurable properties and a means of demonstrating how the overall system goals and objectives are met. 8. Derive a set of stakeholder requirements relative to the system-of-interest needs analysis. 9. Capture any unknowns, risks and assumptions. Derive a method to manage these entities through the product life cycle. 10. Review results with internal stakeholders including appropriate internal development, manufacturing and management teams. 11. Review results with the customer and other appropriate external stakeholders. **Post Conditions -**1. A domain level model is available including all stakeholders, appropriate mission level use cases, the required needs of the stakeholders, the system-of-interest's boundaries and its key interfaces to other entities. References ISO/IEC 15288-2008 - 6.4.1 Stakeholder Requirements Definition Process SEBok - Concept Definition

* + 1. **Derive System Requirements**

**Goal -** The goal of this workflow use case is to derive a set of system level requirements for the system-of-interest based on the all stakeholder's needs requirements. **Primary Actor -** System Architect**Secondary Actors -Preconditions -** "Analyze customer needs" use case is near completion.**Activity -**The following is a set of tasks that are typically captured within a SysML model. 1. Overall guidance items: a. These tasks are not necessarily performed in this order shown. Many are performed simultaneously depending on the needs. Also each task may not be run to full completion before advancing another parallel task. b. It may be determined that some of the tasks or parts of the tasks are not necessary for a particular effort. What tasks are done and to what degree should be determined during the project planning stages. c. As this activity progresses it may be determined that it would be beneficial to initiate another development use cases before completing this use case. This is typically done to provide more insight or to reduce project risks. 2. Identify a reference to the appropriate external documentation and reference (or import) the mission behavior, domain structure and stakeholder requirements from the domain level model. 3. Define a system-of-interest context diagram with the goal of discovering the system interfaces and the system boundaries. This includes identifying external systems, organizations and human actors that may interact or influence the system directly or indirectly.a. Identify all system-of-interest interfacesb. Identify system-of-interest actors 4. Domain level Mission use cases from the domain model may be referenced or enhanced to aid in defining and refining these interactions and the goals of the system. 5. Define a system-of-interest states and modes state machine. 6. From the domain level requirements, analyzes the non-behavioral stakeholder's needs and derive a set of system level requirements for the system-of-interest. a. Capture and demonstrate traceability of these derived system level requirements to the domain level requirements. 7. Analyze system-of-interest behavioral needs - a. By examining the domain level mission use cases derive and capture a set of system level uses cases, their primary actor and the goal of the primary actor. b. For each use case analyze and capture the blackbox behavior of the system required to achieve each use case goal. 8. By examining the system level use cases derive a set of system level requirements. a. These requirements should include functional requirements, external interface requirements, safety requirements, security requirements, environmental requirements, states and modes, design constraints, construction constraints, etc. b. Capture and demonstrate traceability of the derived system level requirements to the domain level requirements and from the system level use cases to the mission level use cases. 9. System validation and verification - a. Derive a set of test cases to validate the system meets the customer's expectations and that all the requirements have been verified. b. Derive the necessary test system architecture, test components and test requirements required to verify and validate the system c. Demonstrate and capture traceability from the test cases to the system level requirements. d. Create a Test Plane. Create and capture a Requirements Verification Matrix10. Capture any unknowns, risks and assumptions. Derive a method to manage these entities through the product life cycle. 11. Review results with internal stakeholders including appropriate internal development, manufacturing and management teams. 12. Review results with the customer and other appropriate external stakeholders. **Post Conditions -** Artifacts created include a system-of-interest context diagram, a set of system level use cases, system level requirements, verification and validation plan, requirement verification matrix. A System Requirement Specification may also be produced including many of the captured artifacts.

* + 1. **Analyze System Life-cycle Costs**

**Goal -** The goal of this workflow use case is to track projected system life-cycle costs starting early at the conceptual phases and continuing through all the remaining product life-cycle stages. **Primary Actor -** Analyst**Secondary Actors -Preconditions -Activity -**This activity includes: Deriving the costs of interest associate with this product Projecting and analyzing these costs through the life-cycle Update and re-analyze as refinements and changes occur through development, manufacturing and support phases of the product.**Post Conditions -**

* + 1. **Workflow Use Case Activity Diagrams**
			1. ***Plan a Response to a Customer Request***



**Figure 5: Plan a Response to a Customer Request**

* + - 1. ***Analyze Stakeholders Needs***



**Figure 6: Analyze Stakeholders Needs**

* 1. **System Development Stage Workflow Use Cases**
		1. **Management Workflow Use Cases**



**Figure 7: Management Workflow Use Cases**

* + - 1. ***Plan a Development Cycle***

**Goal -** The goal of this workflow use case to perform the planning associated with a complete development cycle or smaller individual development cycle iteration. **Primary Actor -** Engineering Manager**Secondary Actors -Preconditions -Activity -**The tasks include:1. Identify program stakeholders and their needs2. Define the development tasks3. Identify Processes/Methods to use4. Allocate the appropriate roles and people to accomplish the tasks5. Evaluate the development environment to determine if the correct set of tools and the revision level of those tools are appropriate6. Schedule tasks7. Identify the metrics to be collected to measure actual costs, progress, quality and complexity8. Review plan with all stakeholders. Post Conditions

* + - 1. ***Manage Development Progress***

**Goal -** The goal of this workflow use case is to manage the planned development efforts to ensure it is manage to successful completion on time and within budget constraints. **Primary Actor -** Engineering Manager**Secondary Actors -** Systems Engineers**Preconditions -** A Development Plan exists**Activity -**This use case can occur at any time during the development cycle however, it must be execute at least at every development iteration and program milestone. The intent is to detect and react to issues as early as possible. 1. Collect metric data associated with progress and costs2. Measure these results against the Development Plan.3. Adjust the Development Plan appropriately4. Publish and review new plan with stakeholders**Post Conditions -**

* + - 1. ***Manage Development Environment***

**Goal -** The goal of this workflow use case is to manage and maintain the integrity of the SE Development Environment. **Primary Actor -** Engineering Manager**Secondary Actors -Preconditions -Activity -**This activity consists of a number of simultaneous monitoring activities to evaluate the integrity of the SE Development environment and to take action when required. These activities can include:1. Manage user access - This activity defines and manages the access groups and assigns users within each group their appropriate access privileges, e.g. read and write access. 2. Manage Tool Upgrades - The goals of this activity are to plan and manage the execution of a tool upgrade within the SE Development Environment while minimizing the impact on the development team.a. Evaluate the compatibility of the new release of the tool with other existing tools and the development environment infrastructure. b. Determine what other tools and/or scripts need to be upgraded to be compatible with the new upgraded tool. c. Evaluate the new features available and ensure they are compatible with existing domain process guidelines and polices. Update guidelines and policies where appropriate. d. Ensure the license agreement for the new release is valid.e. Identify how many copies need to be installed and where.f. Develop a transition plan to allow to users to migrate to the new release at a convenient point in time. g. Notify users of the plan and impact to their assigned tasks. h. When no longer required, remove the existing release. 3. Validate SE Repository Integrity - The goal of this activity is to ensure the integrity of the development repository information and artifacts. a. The development repository includes the information contained in one or more System Architectural Models, other domain specific information and models, and any links established between these items. b. As the development progress the maturity level of this information will increase. This maturity level will affect the extent of the validation performed. c. All SysML based models should be validated against the guidelines/standards established within a company, organization and/or program. The established guidelines should be consistent with industry standards as much as practical and should ensure similar techniques and "look and feel" across all SysML models contained within a program. d. The integrity of links should be measured to ensure completeness, direction of the links and the correct usage of links. e. The intent is to ensure the quality and completeness of all controlled project information is maintained at the highest level possible. f. This activity should be conducted on a regular periodic base. g. When problems are found they should addressed as soon as possible. 4. Synchronize Requirements - The goal of this activity is to ensure that all copies of the requirements within the SE Development System are periodically synchronized to the most appropriate revision level. a. Background - If the development environment maintains multiple copies of requirement data there will be a need to periodically synchronize the multiple copies throughout the development iteration cycle and when baselines occur. The frequency of this will vary depending on the process used, for example these changes may be propagated immediately as they occur, nightly or on planned events. Synchronization maybe required when requirement changes occur from a different abstraction layer or within the same abstraction layer. b. Capabilities -i. Identify the master and copies of each requirement set. ii. Provide a synchronization mechanism, e.g. transfer, OSLC linking, for each requirement set. iii. Configure the system to update the various copies at a pre-arranged event or time. iv. Provide an easy and quick mechanism to allow an SE user that has received an update to select each requirement and measure the change impact on the model. v. If there are no further impacts to the model, accept the change, one at a time or by selecting a group.vi. If the requirement change impacts the architecture, follow the predefine process to return the rejected change to the change source, or update the architecture to make model consistent with requirement change. **Post Conditions -**

* + - 1. ***Create a Baseline***

**Goal -**The goal of this workflow use case is to establish a new baseline of system information that represents the latest set of controlled and validated project information.**Primary Actor -** Systems Engineer**Secondary Actors -Preconditions -Activity -**These baselines are usually associated with a milestone, such as an increment in product development or a new release to manufacturing. Typically there are different levels of baselines from a localized baseline within an organization or small project team to a full baseline representing all program level controlled information. **Post Conditions -**

* + 1. **SE Domain Workflow Use Cases**



**Figure 8: SE Domain Workflow Use Cases**



**Figure 9: Specialty Engineering**



**Figure 10: Analysis Workflow Use Cases**

* + - 1. ***Derive Product Architecture***

**Goal -** The goal of this workflow use case is to evaluate the System Requirements and from them derive the most appropriate architecture to satisfy the customer needs. **Primary Actor -** System Architect**Secondary Actors -Preconditions -** The Exploratory and Concept Stage use cases have been executed and a reasonably stable set of system level requirements are available.**Activity -**The following is a set of tasks that are typically captured within a SysML model. 1. Overall guidance items: a. These tasks are not necessarily performed in this order shown. Many are performed simultaneously depending on the needs. Also each task may not be run to full completion before advancing another parallel task. b. It may be determined that some of the tasks or parts of the tasks are not necessary for a particular effort. What tasks are done and to what degree should be determined during the project planning stages. c. As this activity progresses it may be determined that it would be beneficial to initiate another development use cases before completing this use case. This is typically done to provide more insight or to reduce project risks. 2. Reference the set of artifacts produced from the use cases contained in the Exploratory and Conceptual Stage. If Model-based System Engineering was performed during these stagess this work should be a continuation and refinement of that work. 3. Define logical architecture - Derive a set of candidate subsystems and or components and iteratively validate these selections by analyzing the internal system behavior. 4. Show the structural relationships of these components on a Block Definition Diagram. 5. Show the interconnectivity of these components on an Internal Block Diagram. 6. Analyze the internal system behavior - By using the blackbox behavior of the system level uses cases, create and capture equivalent white box behaviors showing subsystem/component level collaboration required for attaining the same use case goals.7. From the allocated behavior shown on the behavioral diagrams, derive a set of subsystem/component level requirements. These derived requirements can be of any type potentially, but they are typically functional, interface, and performance type requirements. 8. Show traceability from the subsystem/component level requirements to the system level requirements. 9. As each use case progresses conduct the appropriate reviews solicit feedback. Adjudicate all issues and update the reviewed artifacts as needed.10. Perform a configuration management baseline for the appropriate artifacts at the completion of each review. 11. Integrate other domain needs in to the solutions as needed including safety, security, RMA, SWaP, human constraints, etc. 12. Synthesizing the logical architecture to one or more physical solutions13. Select a final solution by performing trade studies with candidate solutions 14. Derive component requirement specifications and collaborate with the implementation teams to finalize the component specifications. The form of these requirements can be a traditional document or an electronic version, such as an html document, or a limited view of the existing System level model. 15. Initiate the use case for deriving the Test and Integration plans**Post Conditions -Output -** Typical deliverables for this effort include Subsystem Requirement Specifications, Software Architecture Document, H/W and/or SW Component Specifications, Interface requirement specifications, Interface design specifications, etc.

* + - 1. ***Evaluate System Safety***

**Goal -** The goal of workflow this use case is to address identified safety related hazards. **Primary Actor -Secondary Actors -Preconditions -Activity -**1. Identify Safety Hazards undesired events and their causes2. Identify applicable safety standards3. Conduct analysis to determine the severity level, the probability of occurrence and assess the level of risk. 4. Determine if the analysis results are acceptable for use. a. If it is acceptable, capture these results and show tractability to identified Hazard. b. If not, determine best corrective solution to eliminate or minimize the level of risk. This could be by design and/or by procedure/processc. Update Model and other information base i. Derive Safety related requirements that address Hazards. ii. Show how these requirements are satisfiediii. Show traceability from Hazards to Risk mitigation requirements to system elements satisfying those requirements.5. Verify solution to determine if Hazard has been appropriately addressed. **Post Conditions -**

* + - 1. ***Perform System RMA Engineering***

**Goal -** The goal of this workflow use case is to manage the system's Reliability, Maintainability and Availability (RMA) as it matures though the development cycle and when complete to ensure the customer's RMA needs are satisfied. **Primary Actor -Secondary Actors -Preconditions -Activity -**This use case begins early in the development cycle and continues through the development as the product becomes more mature. 1. Establish RMA Goals from customer needs2. Capture the reliability of the various system components. 3. Perform analysis to determine the system RMA. 4. Refine component reliability numbers as they become available as the system development matures. 5. If the proposed design does not meet the System RMA goals it can be adjusted through various strategies including mitigation of critical failure risks, increasing the fault tolerance of the system, fault avoidance, preventive maintenance, and controlling the environment of the system. 6. Early in the development process there may not be sufficient detail of the design or component reliability. Therefore assumptions and experience fills in those voids. The ability to capture these early assumptions is necessary. **Post Conditions -**

* + - 1. ***Perform System Security Engineering***

**Goal -** The goal of this workflow use case is to incorporate 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 is available2. A list of applicable policy documentation is availableActivityThis 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: a. Information security governance and risk managementb. Access controlc. Cryptographyd. Physical (environmental) securitye. Security architecture and designf. Business continuity and disaster recovery planningg. Telecommunications and network securityh. Application development securityi. Operations securityj. Legal, regulations, investigations, and compliance2. 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 system points of Interface;a. Identify all external interface pointsb. Identify internal interface points of major subsystems such as server farms, sensors, security management, business network, etc.c. Identifying the points of interface may have been completed earlier in a use case such as "Derive Product Architecture". d. 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 issues7. Conduct appropriate reviews within engineering and with the customer8. Capture test cases that validate the security requirements have been reached9. 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 -**

* + - 1. ***Analyze System Performance***

**Goal -** The goal is to analyze one or more performance requirements and provide the results.**Primary Actor -Secondary Actors -Preconditions -Activity -**This activity starts by defining a specification for this analysis, which can be provided within the context of the architecture model.This specification can include defining an approach, the data, constraints, limitations and assumptions.The analysis is setup and executed. The results are captured and provided to the requesting party. **Post Conditions -**

* + - 1. ***Allocate and Manage SWaP***

**Goal -** The goal of this workflow use case is to manage the values for system size, weight and/or power through the develop cycle to ensure they are maintained within the constraint of a system and/or subsystem.**Primary Actor -** Systems Engineer**Secondary Actors -** Mechanical Design Engineers**Preconditions -Activity -**1. Determine from the system MOE requirements for the maximum resource value for size, weight and/or power. 2. For each of these resources track values for the goal, the desired value, the estimate value and the actual value. 3. Allocate and record a portion of the total resource to each of the various parts of the system (subsystem). This allocated value is referred to as the Desired Value. 4. The sum of these allocations can be less than the maximum resource value. This technique allows some portion of the total value to be used as a reserve (headroom) to help manage these values during the development. 5. Allocation recipients should periodically enter their best estimate at that time as the "estimate value" and a confidence level associated with that value. 6. Allocation recipients should update this value and confidence level as more refined information is known. 7. Periodically the responsible SE rolls up the estimated values from the allocations of the system and evaluates how this compares to the system level desired value. 8. If the system level desired values are within an acceptable tolerance, then there is nothing more to do. 9. If not, interaction with all allocation recipients may be required to determine the best course of action to take to resolve the issue. **Post Conditions -**

* + - 1. ***Perform a Trade Study***

**Goal -** The goal of this workflow use case is to evaluate two or more candidate product solutions to determine the impact of each alternative on selected life cycle costs, such as product cost, production costs, performance, support, installation, etc.**Primary Actor -** Systems Engineer**Secondary Actors -** **Preconditions -** **Activity -**This activity supports trade-off studies and analysis and may include the following functions:1. Describe alternative models that may use common modeling elements.2. A reference library may be imported in to reuse predefined components, functions, parametric models, subsystems, systems, etc.3. Capture the criteria for evaluating system alternatives, and determine the weighting used of each within the analysis.4. Capture the effectiveness measures and corresponding optimization function, to assess how well the alternatives satisfy the criteria and weighting.**Post Conditions -**

* + - 1. ***Analyze Behavior Correctness***

**Goal -** The goal is to execute a behavioral diagram to accomplish the following: Ensure the behavior observed agrees with the intended behavior Verify the specified behavior satisfies customer's needs Refine or improve the specified behavior based on observations **Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* + - 1. ***Manage Product Lines***

**Goal -** The goal of this workflow use case is to ...**Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* + - 1. ***Integrate Human Domain Constraints***

**Goal -** The goal of this workflow use case is to address all required HSI (Human System Integration) domains into the design of the system. **Primary Actor -Secondary Actors -Preconditions -Activity -**HSI consists of integrating all HSI domains including manpower loading, personnel, training, human factors engineering, occupational health, habitability, safety, environment and human survivability. 1. Capture HSI architecture and requirements2. Ensure Information content required by the user is available and the performance of the delivery of this information is adequate **Post Conditions -**

* + - 1. ***Perform Environmental Engineering***

**Goal -** The goal of this workflow use case is to consider environmental impacts of the system and to mitigate risks associated with those impacts. **Primary Actor -** Environmental SE**Secondary Actors -Preconditions -Activity -Post Conditions -**

* + - 1. ***Collaborate with Implementation Domain Team***

**Goal -** Goal of this workflow use case is provide an automated capability to effectively share information between Systems Engineering and the component implementation engineering domains, such as the software, electrical, and mechanical domains. **Primary Actor -** Systems Engineer**Secondary Actors -** Component Engineers**Preconditions -Activity -**1. Interact with implementation domain to derive component specification2. Baseline component specification with consideration to other peer and interactive components3. Attend design reviews as needed4. Address issues, concerns, ideas from component engineering**Post Conditions -**

* + - 1. ***Preform EMI Engineering***

**Goal -** The goal of this workflow use case is to ... **Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* + - 1. ***SE Domain Workflow Use Case Activity Diagrams***
				1. Security Engineering Task



**Figure 11: Security Engineering Task**

* + 1. **Validation and Verification Workflow Use Cases**



**Figure 12: Validation and Verification Workflow Use Cases**

* + - 1. ***Develop Verification Plan and Procedures***

**Goal -** Goal of this workflow use case is to provide a plan to ensure the specified design requirements are fulfilled by the system.**Primary Actor -Secondary Actors -Preconditions -Activity -**This use case begins early in the development cycle and continues to iterate through the remaining development cycles as the product matures. 1. Define and communicate a verification strategy2. Define the verification actions (Test Cases)3. Identify the test components and test data required to execute the test actions. 4. Create Test Procedures for the identified Test Actions**Post Conditions -**

* + - 1. ***Develop a System Integration Plan***

**Goal -** The goal of this workflow use case is to develop a plan to incrementally assemble a system that is consistent with the architectural design.**Primary Actor -Secondary Actors -Preconditions -Activity -**1. Define and capture an assembly sequence and strategy that minimizes system integration time, costs, risks and is consistent with the planned availability of each component. 2. Identify and acquire the test components and test data required for each assembly sequence. 3. **Post Conditions -**

* + - 1. ***Execute a Verification Test Procedure***

**Goal -** The goal of this workflow use case is to execute a test procedure and capture the analyzed results. **Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* + - 1. ***Provide V&V Status***

**Goal -** The goal of this workflow use case is to periodically generate information stating the status of the verification and validation tests, including the status of known issues. **Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* 1. **Production Stage Workflow Use Cases**



**Figure 13: Production Stage Workflow Use Cases**

* + 1. **Support Produceability Engineering**

**Goal -** The goal of this workflow use case is to provide support to manufacturing while manufacturing is evaluating the best means to manufacture the product. **Primary Actor -Secondary Actors -Preconditions -Activity -**This activity should start as soon as practical in the development cycle. The activities include:1. Interacting with Manufacturing Engineering to evaluate the manufacturability of the product2. This may require adjustments to requirements and part selection3. Selecting alternative (second source) parts**Post Conditions -**

* 1. **Product and Service Life Management Workflow Use Cases**

This section of SE Workflow Use Cases covers the Utilization, Support and Retirement Stages.



**Figure 14: Product and Service Life Management Stage Workflow Use Cases**

* + 1. **Support Initial Installation**

**Goal -** The goal of this use case is to provide the necessary support for planning, installing and turn-up of the initial system-of-interest at the customer facility or at an agreed upon site. **Primary Actor -** System Engineer**Secondary Actors -Preconditions -Activity -Post Conditions -**

* + 1. **Architect Sustainability System**

**Goal -** The goal of this workflow use case is to architecturally design a system that will provide the necessary sustainability support for the system of interest. **Primary Actor -** System Architect**Secondary Actors -Preconditions -Activity -**If sustainability is required, the goal of this use case is to architecturally design a system required to sustain the operational systems-of-interest. There are many aspects to sustainability including:**Post Conditions -**

* + 1. **Evaluate Change Request**

**Goal -** The goal of this workflow use case is to evaluate a request for change and determine the cost and time to implement this change. **Primary Actor -Secondary Actors -Preconditions -Activity -**1. Determine what product, resource, time and/or cost constraints exist for a solution2. Evaluate the potential impact of the change on the system. Potentially this could have an impact across the lifecycle including documentation, cost, upgrading field units, manufacturing and/or training3. Determine what engineering domains are impacted and solicit input4. Gather potential solutions5. Determine which solution is superior based on constraints. This may require analysis, trade-studies and/or prototyping6. Produce and deliver an assessment report including the proposed solution description and the impact to life-cycle cost impacts7. Conduct a review **Post Conditions -**

* + 1. **Support System Modernization Plan**

**Goal -** The goal of this workflow use case is assess the options for upgrading equipment in the field to more state-of-the-art equipment. **Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* + 1. **Support System Disposal and Retirement**

**Goal -** The goal of this workflow use case is to provide the necessary support for the planning, uninstalling, removal disposal of installed systems. **Primary Actor -** Systems Engineer**Secondary Actors -Preconditions -Activity -Post Conditions -**

* 1. **Supporting Activity Behaviors**
		1. **Measure a Change Impact**



**Figure 15: Measure a Change Impact**

* + 1. **Conduct a Review**



**Figure 16: Conduct a Review**

* + 1. **Import Reference Materiel into Model**



**Figure 17: Import Reference Materiel into Model**

* + 1. **Perform a Trade Study**



**Figure 18: Perform a Trade Study**

1. **References**

INCOSE-TP-2003-002-03.2.2, INCOSE Systems Engineering Handbook v. 3.2.2, October 2011Pyster, A. and D.H. Olwell (eds). 2013. The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 1.2. Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed DATE. www.sebokwiki.org/ International Standard - ISO/IEC 15288 and IEEE 15288 - 2008, Second Edition 2008-02-01, Systems and software engineering - System life cycle processes

1. **Addendum**
	1. **Workflow Use Case Templates**
		1. **Workflow Use Case Pattern Template 1**

**Goal -** The goal of this use case is...**Primary Actor -Secondary Actors -Preconditions -Activity -**This use case begins early in the development cycle and continues to iterate through the remaining development cycles as the product matures. Early in the development process there may not be sufficient design detail to determine if the product needs can be meet with 100% certainty. Therefore assumptions and experience are required to fill those gaps until the design matures. These unknowns, risks and assumptions need to be captured and managed. 1. Establish the customer needs that are impacted by this domain including Measures of Effectiveness 2. Capture information design and assumptions 3. Measure domain design effectiveness, typically by use of analysis 4. Measure impact in other domain areas 5. Refine design 6. If the proposed design does not meet the System goals execute the following steps. 7. Iterate back to the appropriate previous step until design seems sound 8. Conduct appropriate peer/engineering/customer level reviews 9. Capture results of task including defined metrics**Post Conditions –**



Figure 9: General Workflow Pattern 1

* + 1. **Workflow Use Case Pattern Template 2**

**Goal -** The goal of this workflow use case is...**Primary Actor -Secondary Actors -Preconditions -Activity -Post Conditions -**

* 1. **Industry Available Product Phases**



**Figure 19: Industry Available Product Phases**