Introductory Welcome Topics

1/13/2023

Table of Contents

1 What is a System?

Everyone uses the term and has an intuitive notion of what a system is, but there is a formal definition. INCOSE and ISO/IEC/IEEE define a system as:

"A combination of interacting elements organized to achieve one or more stated purposes."

This general definition covers almost everything you can think of including household appliances, transportation management systems, the latest weapon system, even the solar system, all of these are systems.

The two elements of this definition drive many of the processes we use. They are:

- Elements that interact. A system is a sum of parts that must work together.
- Stated purpose. Those elements interact to serve a clearly defined purpose.

Without understanding the purpose, defining what a project must build becomes impossible. But with the purpose defined, those parts and how they are integrated can be measured at every step of procurement from preliminary design to operation.

Reference: ISO/IEC/IEEE 15288-2015, ISO/IEC/IEEE International Standard – Systems and software engineering – System life cycle processes, https://standards.ieee.org/ieee/15288/5673/

2 What is Systems Engineering?

To understand "What is Systems Engineering?" it is useful to discuss what systems engineers do. Systems engineers are concerned about the "big picture" of a project. They oversee all aspects of a project in a variety of fields, such as electrical, civil, transportation and manufacturing. Systems engineers collaborate with project team members to ensure that the parts (e.g., software, hardware, interfaces, security systems, databases, users, etc.) of the project work together to accomplish its stated purpose. Systems engineering is needs focused and requirements driven.

More formally, the systems engineering organization called the International Council on Systems Engineering (INCOSE) defines systems engineering as follows: Systems Engineering *is a transdisciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts, and scientific, technological, and management methods.*

- *Transdisciplinary* - "transcends" all of the disciplines (i.e., fields of study) involved, and organizes the effort around common purpose, shared understanding and "learning together" in the context of real-world problems or themes. Systems engineering is multidisciplinary, focusing on how users get the technologists to meet their needs.
- *Integrative* - involves either interdisciplinary (e.g. integrated product teams) or multi-disciplinary (e.g. joint technical reviews) methods
- *Engineered Systems - are a composite of people, products, services, information, and processes (and possibly natural components) that provides a capability that satisfies a stated customer need or objective.*
- *Systems Principles and Concepts* - are the ways that systems thinking and the systems sciences infuse systems engineering.

Link to a more detailed definition of SE: https://www.incose.org/about-systems-engineering/system-and-se-definition/systems-engineering-definition

Learn more about Systems Engineering at https://www.incose.org/about-systems-engineering .

Learn more about INCOSE at https://www.incose.org/about-incose.

3 Why Use SE?

The overall goals of systems engineering are to reduce risks, identify and correct defects as soon as possible, provide a common language between subject matter experts (SMEs) and system designers/technologists, and a system design that responds to **needs** and **requirements** instead of dictating **needs** and **requirements**

Other benefits of systems engineering include:

- better system documentation
- higher level of user engagement
- system functionality that meets user needs
- potential for shorter project cycles
- systems that can evolve with a minimum of redesign and cost
- higher level of system reuse
- more predictable outcomes from projects

4 When to Use SE?

Systems engineering mitigates the risk of technology-based projects not meeting the needs of their users. Therefore, agencies should use systems engineering approaches when projects demonstrate risk, and the detail applied to the SE process should be commensurate with that risk. State departments of transportation develop procedures in consultation with their FHWA Division office primarily aimed at assessing risk as the basis for determining how much systems engineering the agency should undertake. Agencies should determine and follow these procedures.

Even in low-risk projects, however, agencies should address all the elements of a complete systems engineering process, including user needs and requirements in advance of design, procurement, and implementation, followed by verification and validation steps to demonstration that requirements have been fulfilled and needs have been satisfied. Lower-risk projects, however, may use needs and requirements that have already been documented as part of some prior process, and need less detail in each of these steps. Higher-risk projects may need addition systems engineering activities to fully document needs and requirements in sufficient detail to mitigate identified risks. And the highest-risk projects may need to be conducted in phases to minimize investment until the most serious risks have been addressed.

For More Information:

Several topics in Document View provide additional information:

- US DOT Regulations related to SE
- Agency Implementation of Systems Engineering Processes

5 Key SE Principles

Some of the key systems engineering principles are:

Viewing the system from the stakeholder points of view, this means walking in the shoes of the stakeholders, with more emphasis on the system's users and owner(s). Key processes for this principle include needs assessment, user need elicitation, developing a Concept of Operations, and especially user and all stakeholder involvement.

Start at the finish line defines the expectations for the system and the way the system is going to operate. The details might change but the key concepts and ideas on what the system should do to meet particular user needs should remain consistent. Key processes for this principle include Concept of Operations and Validation Plan.

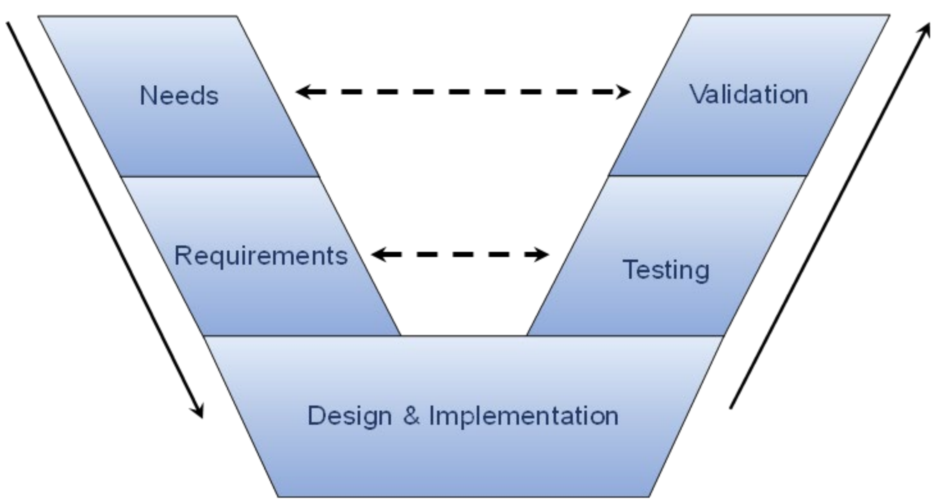
Address risks as early as possible when the cost impacts to addressing those risks are the lowest. Key processes for this principle include risk management, requirements, and stakeholder involvement.

Push technology choices to the last possible moment. Define *what* is to be done before defining *how* it is to be done.

Focus on interfaces of the system during the definition of the system. Defining clear and standard interfaces and managing them through the development will ease the integration of the individual elements of the system.

Understand the organization of the system's owner, stakeholders, and development team.

6 High-level Systems Engineering Overview



As shown here, the systems engineering process can be shown using the letter "V". Let's start at the upper left of this simplified Vee diagram. The first major process step is capturing the needs of the users. These user needs provide the basis for forming the requirements in the next step. The requirements inform the bottom of the Vee Diagram depicting Design and

Implementation. This step is where the system gets built. Once the system is built, we travel up the right side of the Vee diagram where we test the system. The last step on this diagram is Validation where we make sure that the system is useful to the users.

Time increases from left to right as depicted by the solid arrows on the outside of the Vee diagram. The dashed arrows in the middle of the Vee indicate that the Testing is against the requirements and the Validation is ensuring the needs are being met by the final system.

For More Information:

The Process View on this website is built around a slightly more complex Vee diagram that adds a few steps and includes "wings" that put the SE development process in a broader context. The following Process View topic provides an overview of the complete Vee diagram:

- Vee Overview

7 Know Your Needs

Leveraging one of the key systems engineering principles, namely "***Viewing the system from the stakeholder points of view",*** it is critical that the ITS system be developed with a fixation on the expectations and needs of the users. User expectations include needs, desires, capabilities, and constraints that are expressed by those who have a vested interest in the system. Initial system development must focus on user needs including how to identify them through elicitation. There are many different elicitation techniques that can be used including interviews, scenarios, prototypes, facilitated meetings, surveys, and observations. Each of these techniques can be used in combination to discover the user's needs.

Concept of Operations (ConOps) - A document describing how a system operates during the life cycle phases to meet user expectations. It describes the system characteristics from an operational point of view, stimulating the development of requirements and architecture as they relate to users of the system. The ConOps provides the basis for system validation.

For More Information:

The Concept of Operations topics in Process View and Deliverable View cover needs definition and documentation.

- Concept of Operations Process
- Concept of Operations Template

8 Developing Requirements

Requirements are central to the entire systems engineering process. The concept of operations details including user needs and operational scenarios, inform and spawn detailed requirements.

Requirements emerge from the user needs. Requirements are used to guide the design and determine that the design is complete and correct. Later we will use the requirements after system implementation to demonstrate the system functions correctly. The requirements are referenced during the testing stages and the system acceptance is predicated on the testing against the requirements being successful. Well written requirements are necessary, if not, the end system may not meet the user needs and expectations. The requirements also serve as the basis for procurement from selection to final acceptance.

Included in this topic is the need to have well formed requirements, namely are they:

- Necessary
- Clear
- Complete
- Correct
- Feasible
- Verifiable

It is critical to document your requirements. They need to be sequentially numbered and frequently in a hierarchy so they can be referred to in traceability tables.

For More Information:

Visit Process View and Deliverable View for more information on the process of developing requirements and documenting them:

- Requirements Process
- Requirements Template

9 Validation vs. Verification

The terms "Validation" and "Verification" sound very similar. In systems engineering they are used in a distinct manner. The definitions for both are found below:

Verification - The process of demonstrating that a system fulfills specifications and requirements. It answers the question: Did we build the system correctly?

Validation - The process of demonstrating that a system provides the value intended (capability) and meets the original needs defined in the ConOps. It answers the question: Did we build the right system?

In almost every case, verification of the system comes first followed by validation by the users. You may encounter the term "in-process validation", this concept validates the stages in the life cycle process as they are completed with the other completed stages.

For More Information:

Visit Process View for more information on the Verification and Validation processes:

- Verification Process
- Validation Process

10 SE Life Cycle Models

You may not be familiar with the term "life cycle model". A life cycle model describes the distinct stages of a system's "life". Generally, a system moves through different stages, from planning, concept, development, implementation, operations and support with final retirement. The role of the systems engineer encompasses the entire life cycle of the system. Within the context of the system life cycle stages described above, there are several life cycle approaches used by agencies to develop a technology project. The best development strategy depends on how much you know about the system that you want to implement, whether you have all the funds that you need to implement the system in one fell swoop, your agency and contractor capabilities, and your assessment of the project risks. Several of these life cycle approaches include:

Sequential Methods – Sequential methods are characterized by a system moving through a set of defined processes with gates that are passed through between each process. These gates are usually deliverables such as a Concept of Operations or Requirements document. Traceability between the processes is another trait of a sequential method. Examples of sequential methods are the traditional "waterfall" transportation project development model as well as the Vee model.

Incremental Methods - The most common incremental method is really a variation of a sequential method. It may have incremental aspects relating to design and development of the system, but a key aspect of this approach is that the complete system is initially planned and specified. In this case, you are making one pass through the first part of the development process to determine the needs addressed and the requirements of the system. One or several projects then iterate through the latter part of the development process for each phased increment.

Iterative Methods - There are several development methods that employ iterative approaches throughout the development process. In these methods, developers plan, specify, and implement an initial system capability. Following the initial development, which may or may not be determined acceptable for operational use, this process leverages experience gained with the initial system to define the next iteration to fix problems and extend capabilities.

These iterative approaches are often used when the requirements are unclear from the beginning, or the stakeholder wishes to hold the system of interest open to the possibilities of inserting new technology or capabilities

One type of iterative method, described under the term Agile development, can be considered for use for ITS projects. Agile is a technical development strategy characterized by frequent, incremental deliveries of value. The requirements and architecture for a system of interest are allowed to evolve as development progresses, rather than attempting to fully define them at the beginning of development.

One of the most common iterative methods is referred to as the Scrum method. Scrum is an iterative agile methodology for managing product development within which people can address complex adaptive problems, while productively and creatively delivering products of high value. The original idea of the agile concept was to provide an alternative to the sequential waterfall method for software development, providing an alternative to documentation driven, heavyweight software development processes. The idea was to incrementally deliver planned functionality earlier in the development cycle.

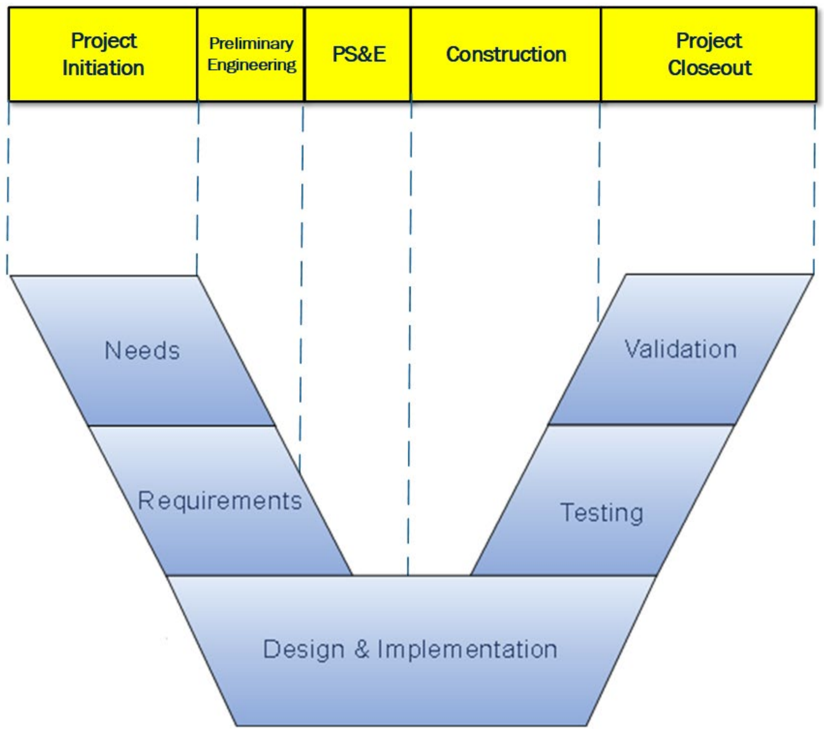
For More Information:

Document View provides some additional context and detail for SE Lifecycle Models:

- Systems Engineering Life Cycle Models

11 Traditional Vs. Vee Life Cycles

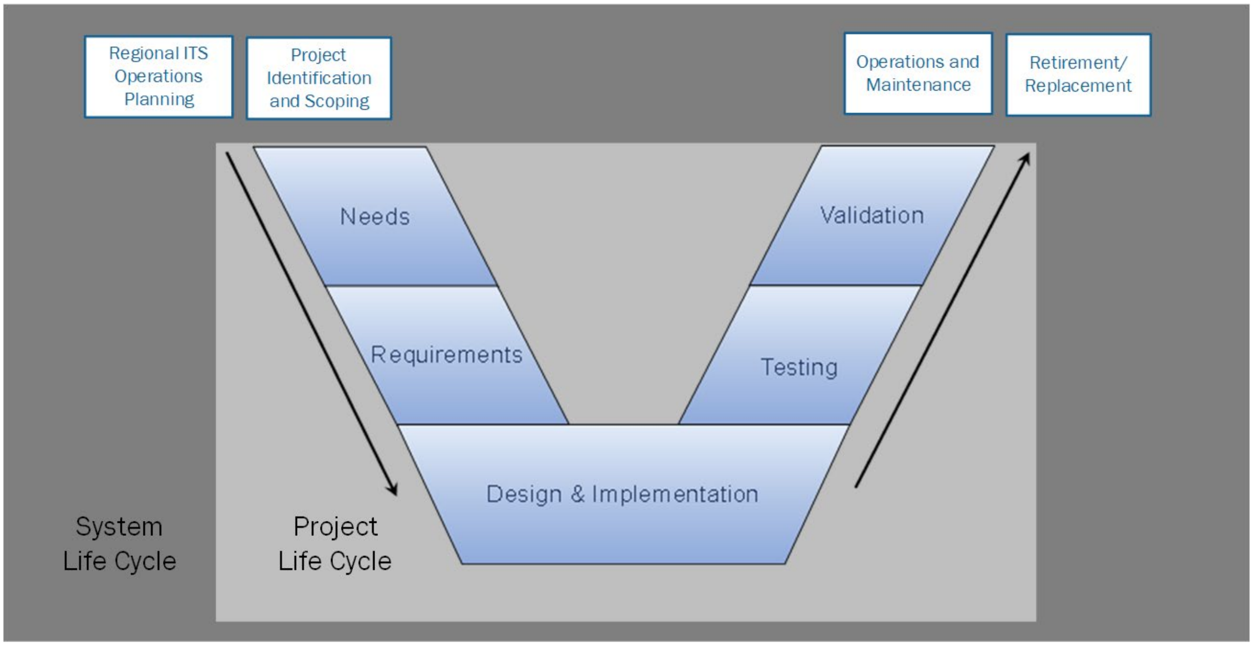
The two most common life cycle models are the traditional transportation life cycle model and the Vee life cycle model. While these two models may look quite different, you can easily relate the steps of the traditional transportation life cycle model (in yellow below) to the Vee life cycle model (simplified version in blue).



Dashed lines from the traditional process to the Vee process show how the two processes relate to each other. Generally, Project Initiation through Project Closeout is related to the Vee Needs through Validation processes. There is not quite a one-to-one relationship between the different stages or phases of the models, but since each are versions of sequential models, they line up fairly well.

12 System Vs. Project Life Cycles

When discussing a life cycle, you can distinguish between a system life cycle and a project life cycle. The project life cycle encompasses all the activities of a project, how a project is planned, designed, developed, and tested and covers from inception to completion for a specific project. The project life cycle is a subset of the larger system life cycle which begins with the operations planning that leads to the need for a system all the way through to the operation, update, and finally retirement or replacement of the system. The diagram below uses the simplified Vee model to illustrate the difference between a project life cycle and the overall life cycle of a system. Note that a system may have one or more projects that are implemented to form the entire system.



13 SERF, SEMP and PMP

At the outset of a project, it is important to determine the systems engineering tasks that will be performed and document those tasks as part of the project development process. The various documents surrounding the systems engineering process may have various titles. These titles are in common use within the transportation industry although they may not have the same meaning among agencies.

13.1 Systems Engineering Review Form

Typically, the Systems Engineering Review Form (SERF) is a form that is used by agencies to determine how much systems engineering is required. It includes project information, risk assessment and a project Systems Engineering Analysis (SEA) which, at a minimum, should satisfy the 23CFR 940.11 Federal regulations. A SERF can be tailored based on the agency's needs.

13.2 Systems Engineering Management Plan

The Systems Engineering Management Plan (SEMP) is a document that describes the technical approach on how the processes and activities of the systems engineering life cycle stages will be managed and conducted. For high-risk projects with substantial systems engineering effort, a separate SEMP may be warranted. In many cases, the systems engineering tasks may be identified as part of the Project Management Plan (PMP).

13.3 Project Management Plan

A Project Management Plan (PMP) is usually a formal document that helps outline all the components of a project. This includes stakeholders and scope all the way to risk management and contingency plans. Far more than just a schedule, this plan documents how the project will be managed and achieve success.

For More Information:

Document view provides more detailed discussion of the SERFs used by states. Deliverable View provides templates for both the SEMP and the PMP. The Process View describes the Project Planning process step when these documents are created and the Project Management process that uses and refines these documents over the course of the project lifecycle.

- Agency Implementation of Systems Engineering Processes (for a discussion of SERFs)
- Systems Engineering Management Plan Template
- Project Management Plan Template
- Project Planning Process
- Project Management Process

14 SE Training

Below is a list of organizations providing systems engineering training focused on transportation. The only general systems engineering resource is INCOSE.

National Highway Institute (NHI) – refer to https://www.nhi.fhwa.dot.gov/home.aspx, search on "systems engineering", there are both Instructor-led Training (ILT) and Web-based Training (WBT).

Professional Capacity Building (PCB) ITS JPO Program – refer to https://www.pcb.its.dot.gov/default.aspx, search PCB website for "systems engineering".

Consortium for innovative Transportation Education (CITE) – refer to https://www.citeconsortium.org/ , search CITE website for "systems engineering".

U.S. DOT ARC-IT National ITS Reference Architecture – refer to https://www.arc-it.net/html/resources/training.html, delivery options range from convenient web-based training for individuals to facilitated workshops for one or more regions.

International Council on Systems Engineering (INCOSE) – refer to https://www.incose.org/. INCOSE is a not-for-profit membership organization founded to develop and disseminate the interdisciplinary principles and practices that enable the realization of successful systems. INCOSE is designed to connect SE professionals with educational, networking, and career-advancement opportunities in the interest of developing the global community of systems engineers and systems approaches to problems.

15 SE Regulations

Federal regulations (reference 23 CFR 940) that are specifically related to ITS systems engineering can be summarized with definitions and requirements. The definition of ITS is "Electronics, communications, or information processing, used singly or in combination to improve the efficiency or safety of a surface transportation system." An ITS Project is defined as "Any project that in whole or in part funds the acquisition of technologies or systems of technologies that provide or significantly contribute to the provision of one or more ITS user services as defined in the National ITS Architecture."

The regulations require the development of regional ITS architectures that can be used to plan the integration ITS deployments within a region and documentation of systems engineering processes relating to ITS projects.

On July 7, 2022, a memorandum was published implementing Section 11304 of the Bipartisan Infrastructure Law (BIL), enacted as the Infrastructure Investment and Jobs Act (Public Law 117-58), which directs the Secretary of Transportation to develop guidance for using existing flexibilities with respect to Systems Engineering Analyses described in 23 CFR part 940.

15.1 Regional ITS Architectures

A regional ITS architecture is a framework for institutional agreement and technical integration for the region. The regional ITS architecture defines the links between the pieces of the system and the information that is exchanged on each connection. Over 300 regional ITS architectures have been developed so chances are good that you already have one in your region.

The federal regulations (reference 23CFR 940.9) require regional ITS architectures to include the following:

- Description of the region
- Stakeholders
- Operational concept (Roles & Responsibilities)
- System functional requirements
- Interface requirements
- ITS standards
- Agreements
- Sequence of projects
- Maintenance Plan

The Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT) is used as a template to create regional ITS architectures that are tailored for a specific state, metropolitan area, or other region of interest (e.g., a major corridor or a National Park). ARC-IT provides the fundamental building blocks: the physical objects (subsystems and terminators), interfaces (as defined by the information flows), service packages, functional objects, and functional requirements that are selectively included in the regional ITS architecture and customized as necessary to fully reflect the envisioned regional transportation system.

The Regional Architecture Development for Intelligent Transportation (RAD-IT) software tool provides an easy way to personalize and customize ARC-IT for a specific region.

15.2 Systems Engineering Analysis (SEA)

The federal regulations (reference 23CFR 940.11(c)) require the documentation of system engineering processes relating to ITS projects. The regulations define a systems engineering analysis or SEA for an ITS project as the following:

1. Identification of portions of the regional ITS architecture being implemented
2. Identification of participating agencies roles and responsibilities
3. Requirements definitions
4. Analysis of alternative system configurations and technology options to meet requirements
5. Procurement options
6. Identification of applicable ITS standards and testing procedures
7. Procedures and resources necessary for operations and management of the system

Although the language of the federal regulations states "systems engineering analysis", it could be better thought of as a "systems engineering approach" or "systems engineering processes".

The Systems Engineering Tool for Intelligent Transportation (SET-IT) software tool provides extensive drawing capabilities to produce project architectures and support Concept of Operations and Requirements definition. SET-IT content is based on ARC-IT and is compatible with the RAD-IT software tool.

15.3 Intelligent Transportation Systems Projects: Flexibilities with Systems Engineering Analysis Memorandum

This memorandum implements Section 11304 of the Bipartisan Infrastructure Law (BIL), enacted as the Infrastructure Investment and Jobs Act (Public Law 117-58), which directs the Secretary of Transportation to develop guidance for using existing flexibilities with respect to Systems Engineering Analyses described in 23 CFR part 940. The goal is to avoid unnecessary burden on State and local governments, prevent unintentionally imposing requirements exceeding those outlined in law and regulation, and enhance the necessary engagement and collaboration between division staff and their counterparts in State and local agencies with respect to ITS projects. This memorandum may be found here: Information Memo - Systems Engineering for ITS Projects.

For More Information:

Document View includes a similar overview of the Systems Engineering Analysis requirements with some additional detail:

- US DOT Regulations