

Module 2: Systems Engineering

Authored by **Bruce Eisenhart**, Vice President of Operations, Consensus Systems Technologies (ConSysTec), Centreville, VA, USA

Purpose

The purpose of this module is to provide an overview of the systems engineering process (SEP) that is central to the development of intelligent transportation systems (ITS) projects. The SEP is an interdisciplinary, structured process that meets the needs of the users, providers, and other stakeholders while maintaining the project schedule and budget. This module also provides an overview of related topics, including the National ITS Architecture and other architectures, the ITS Standards Program, and the use of the SEP for planning and deploying ITS projects.

Objectives

This module has the following objectives:

- Provide an introduction to the SEP and describe how to apply it to the development of ITS projects.
- Provide an overview of ITS architectures.
- Review the role of ITS standards in the development process.
- Discuss the Architecture and Standards Rule and the role of systems engineering and ITS architecture in addressing the requirements of the rule.
- Identify how the SEP supports transportation planning.

Introduction

Systems engineering offers a framework for developing complex systems used in engineering projects. In a retrospective on the National Aeronautics and Space Administration's (NASA) Project Apollo, systems engineering is credited with the project's success in the 1960s. "NASA personnel employed the 'program management' concept that centralized authority and emphasized systems engineering. The systems management of the program was critical to Apollo's success. Understanding the management of complex structures for the successful completion of a multifarious task was a critical outgrowth of the Apollo effort."¹

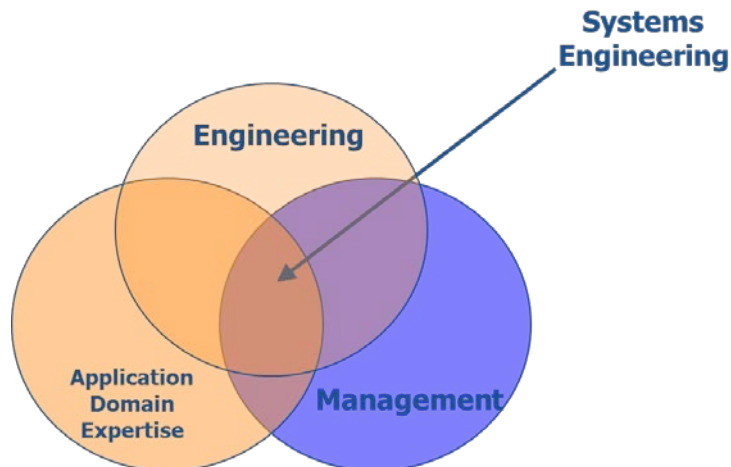
Systems engineering provides a structure for deriving order from chaos, as demonstrated in this amusing video: www.youtube.com/watch?v=t0tGd5igKZM&list=TL2cvF2Qi7Bc4. This module provides an introduction to the SEP and its use in the planning and development of ITS projects.

The International Council on Systems Engineering (INCOSE) gives the following short definition of the field:

Systems engineering is an interdisciplinary approach and means to enable the realization of successful systems.²

The term *interdisciplinary* is a fundamental concept in systems engineering. To successfully develop a system requires the application of basic engineering discipline, management discipline, and expertise in the application domain of the system. In the case of ITS, that application domain is transportation engineering. The intersection of these disciplines is the realm of systems engineering.

Figure 1. Systems Engineering Improvement



Source: U.S. Department of Transportation’s Research and Innovative Technology Administration (RITA), National ITS Architecture SE Process Improvement Presentation.

The goal of systems engineering is to successfully develop a system. Although there are many measures of success, the goal is to create a quality system that is developed on budget and on schedule and meets the technical requirements and expectation of the stakeholders who will use or maintain the system.

Systems engineering as an idea dates back to the 1930s. The basic principles and processes were developed in the 1940s and 1950s to support the development of increasingly complex military systems. In the past 20 years, the use of the SEP in more generalized systems development has been codified in a set of international standards (see the Additional Resources section). In the case of ITS project development, the use of standards is relevant because the systems aspect of ITS is increasingly complex, combining hardware, software, and communications to provide transportation services.

The U.S. Department of Transportation (USDOT) recognized the potential benefits of the systems engineering approach for ITS projects. The Federal Highway Administration (FHWA) Rule 940 and Federal Transit Administration (FTA) Policy,³ which were enacted in 2001, require that a systems engineering analysis (SEA) be performed for ITS projects that use funds from the Highway Trust Fund, including the Mass Transit Account. In addition, the FHWA rule requires the development of a regional ITS architecture, defined by the rule as “a framework for

ensuring institutional agreement and technical integration for the implementation of ITS projects in a particular region.” This module expands on both of these topics.

Key Concepts of the SEP

The systems engineering process is defined by a set of key concepts that will be further developed in the step-by-step review of the process below. These key concepts involve the following approaches:

- Consider the entire life cycle of a system. The SEP does not just concern itself with the timeline of a project’s start to completion. The SEP considers planning activities that occur before the project starts and covers steps following initial deployment, including operations and maintenance and system retirement or replacement.
- Focus on stakeholder involvement. The SEP involves stakeholders in all steps in the process, from identifying the needs through validating the system. Stakeholder involvement is a key part of the successful application of the SEP.
- Understand the problem to be addressed. The SEP views the problem from the stakeholders’ point of view by defining needs from the user’s perspective.
- Address project risks as early as possible. The earlier a risk is identified and resolved, the less the impact will be on the project’s cost or schedule.
- Clearly document the process and the output of each step. Stakeholders must review and document the outputs of each step in the process and make informed decisions about proceeding to the next step.
- Tailor the SEP to fit the needs of the project. The SEP is scalable based on the size, complexity, and risk of a project; it is not a one-size-fits-all process.

Benefits and Costs of Using the SEP

The systems engineering process can improve the quality of the products created by an ITS project, reduce the risk of schedule and cost overruns, and increase the likelihood that the implementation will meet the user’s needs. Other benefits include improved stakeholder participation and better documentation of not only the end products but also the development process itself. Anyone who has tried to pick up a project midstream after a key person unexpectedly left will understand the benefit of good documentation during the development effort. In addition, the benefits of using the SEP can continue throughout the life cycle of the system, reducing operational costs by implementing the product correctly the first time and also reducing the need for later modifications.

The SEP provides a number of benefits as listed above, but these benefits do come with some reapportioning of development costs. In general, the use of the SEP will result in more effort expended at the early stages of the development process—those steps leading up to detailed design. Additional costs may also be incurred during the systematic verification described below. But these planned costs can significantly reduce the likelihood of cost overruns resulting from redoing work to address problems later in the development process. The amount of effort spent on the SEP depends on the size, complexity, and amount of risk associated with the project. Studies conducted in the broader arena of systems development have found that a

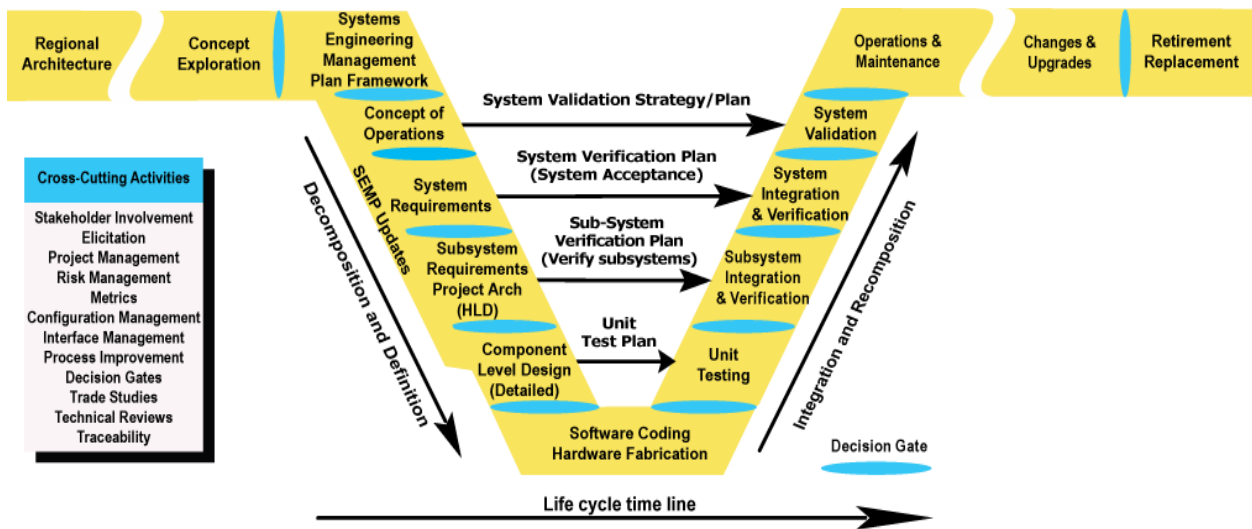
systems engineering effort typically represents about 10 percent of the overall project cost. In general, the greater the size, complexity, or risk associated with the project, the greater the likelihood that the effort associated with the SEP will pay off in achieving the benefits listed above.

Overview of the Systems Engineering Process

The following description of the systems engineering process for ITS is based on two key documents that have been developed by FHWA: *Systems Engineering for ITS—An Introduction for Transportation Professionals*,⁴ and *Systems Engineering Guidebook for ITS*.⁵

Many different process models have been developed over the years that specify a series of steps that make up the systems engineering approach. The V process model illustrated below has gained wide acceptance in the standards community and is the model chosen by USDOT in the two references above. One of the key reasons for the choice of this model is that it illustrates some key SEP principles about the relationship of the early phases of project development to the results of the project. The following discussion will walk through each step in the process. This V process model is taken from the *Systems Engineering Guidebook*, which has a hyperlinked website with considerable additional detail about each step in the process. Hyperlinks to the relevant section of the guidebook are accessed by clicking on the process step icon. As the discussion proceeds from one step to the next, the steps, particularly on the left side of the V, will usually be iterative in nature.

Figure 2. V Process Model



Regional ITS Architecture

The first step in the process is to consider how the system to be developed is described in the regional ITS architecture, which represents a plan for the deployment of ITS in the region. As a result of the requirements of 23 CFR 940 (discussed in the ITS Architecture section later in this module), over 400 regional ITS



architectures have been developed throughout the United States. These regional architectures identify the stakeholders in the region, the systems the region has implemented (or plans to implement), and the ITS services currently deployed or planned for deployment in the region. As part of these regional ITS architectures, the ITS projects planned for the region are described, and in most architectures the ITS projects are mapped to some portion of the architecture. The purpose of this initial step is to put a project into the context of the region's overall ITS plan.

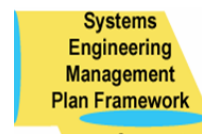
Concept Exploration

The concept exploration step in the process is used to perform any initial feasibility analysis, benefits analysis, or needs assessment required to facilitate the planning of the system. This results in a business case and specific cost-benefit analyses for alternative project concepts. The output of this step can be a definition of the problem space; key technical metrics of the system; and refinements to the needs, goals, objectives, and vision. This step provides the justification for the project to move forward into development. The activity may result in combining or dividing candidate projects based on the best cost-benefit analysis. This step has the first of many "decision gates," which show as the light blue ovals between steps in the V diagram. These decision gates represent key points in the process where a specific decision is made, based on documented outputs or possibly technical reviews, to advance to the next step in the process. The decision gate in this step is to gain management support and approval for the project to move into the programming of funds phase of the process (where funds are allocated for the development of the system). This step is often undertaken when some significant new initiative or system is envisioned. In the case of many ITS projects, particularly those that involve upgrades to existing systems, this step may not be needed or might be abbreviated.



Systems Engineering Management Planning

Each project that moves forward into development must be planned. Planning takes place in two parts. In part one, the system's owner develops a set of master plans and schedules that identify what plans are needed and, at a higher organizational level, the schedule for implementation of the project. This part becomes the framework for what is developed in part two. The plans are completed during the following steps in the SEP (from the concept of operations to detailed design). These plans, once approved by the system's owner, become the control documents for completion of the development and implementation of the project. The key output of this step is the Systems Engineering Management Plan (SEMP) document.



The SEMP is an extension of the classical project management plan, but it controls the technical development of an ITS project. It is the top-level plan for managing the systems engineering effort, defining how the systems engineering portion of the project will be organized, structured, and conducted and how the total engineering process will be controlled to provide a product that fulfills customer requirements. This SEMP is the first of a series of documents that are created as part of the SEP to document the project and the process under which it is being developed. These documents will be described in each of the subsequent steps and represent

one of the key concepts of the SEP—that is, clearly documenting the process as well as the outputs of each step of the process.

The SEMP identifies all the required documents and plans that will need to be developed throughout the project. The initial version of the SEMP (sometimes referred to as the SEMP Framework) is developed at the beginning of the project and is usually not a large or complex document, because it is basically an outline of what is to come. Although the SEMP is a fairly new concept in ITS project development, it is an extremely important tool for managing the technical side of the project. The SEMP is intended to guide the agency or owner of the project in the management of the project. It is not meant to be a document written by the contractors on a project for their own internal management; rather it is the primary tool for the project owner to monitor contractor work and progress.

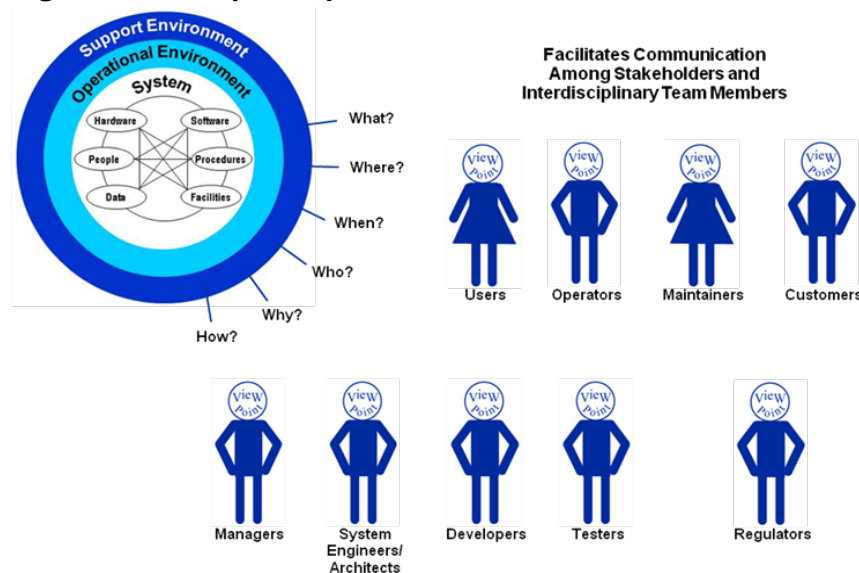
An excellent template for creation of the SEMP (as well as examples of SEMP created on actual projects) is given in the *Systems Engineering Guidebook for ITS*. The guidebook contains templates and examples for all of the systems engineering documentation discussed in the steps below.

Concept of Operations

The purpose of the concept of operations (ConOps) step is to clearly convey a high-level view of the system to be developed that each stakeholder can understand. This foundation document frames the overall system and sets the technical course for the project. A good concept of operations answers who, what, where, when, why, and how questions about the system from the viewpoint of each stakeholder, as shown in the figure.



Figure 3. Concept of Operations



Source: ANSI/AIAA G-043A-2012.⁶ (Reprinted with permission from the American Institute of Aeronautics and Astronautics.)

The ConOps is one of the first key outputs of the SEP and forms the basis for the definition of requirements in the next step. The ConOps stage of the SEP is used to ensure that the system developers' documentation shows a thorough understanding of the users' needs.

This step embodies several of the key concepts of the SEP, which include (a) engaging the stakeholders, whose input is key to the development of the ConOps; (b) focusing on the problem and on user needs; and (c) beginning with the end in mind, that is, defining key system outputs as well as key operations and maintenance considerations.

To accomplish these steps, the ConOps documents the way the envisioned system is to operate and how the envisioned system will meet the needs and expectations of the stakeholders. At this step the SEP V diagram shows the first of several arrows leading from the left side of the V (the decomposition and definition side) to the right side of the V (the integration side), indicating the creation of a System Validation Strategy or System Validation Plan. The ConOps defines the user needs, and at this step, the SEP owners may create a plan for the validation of those needs, which will be used to support the system validation step described below.

System Requirements

One of the most important attributes of a successful project is a clear statement of requirements that meet the stakeholders' needs. EIA-632 defines a requirement as "something that governs what, how well, and under what conditions a product will achieve a given purpose."⁷ This is a



good definition because it touches on the different types of requirements that must be defined for a project. Functional requirements define *what* the system must do, performance requirements define *how well* the system must perform its functions, and a variety of nonfunctional requirements define *under what conditions* the system must operate.

The requirements step is an important one. Every ITS project should have a documented set of requirements that are approved and for which a baseline is established (using a configuration management process, which is discussed later under cross-cutting activities). Of course, this step does not mean that a new requirements specification must be written from scratch for every project. Projects that enhance or extend an existing system may build on existing system requirements, or in the case of field device deployments, the project may operate using agency specifications for the device.

Involving stakeholders in requirements development is important because they are the foremost experts concerning what functions the system should have, how well the system must perform its functions, and under what conditions the system must operate. The job of the systems engineer is to elicit this information from the stakeholders. The result of this step should be a documented set of requirements that completely address all of the user needs defined in the ConOps. As shown in the V diagram, the system requirements form the basis for creating a System Verification Plan that defines a verification case and method for each requirement. The system requirements also must trace back to the user needs in order to show that all the needs

are being addressed. Traceability is a key concept in the SEP and is discussed in more detail later.

Subsystem Requirements in High-Level Design

The systems engineering approach defines the problem before defining the solution. The previous steps in the V diagram have all focused primarily on defining the problem to be solved. The high-level design step is the first step in defining the project solution. It is an important transitional step that links the requirements that were defined in the previous step with a design framework that will be built upon and implemented in the following steps.⁸



The key outputs of the high-level design step are to break the system into subsystems and to define the key interfaces between the subsystems. One way to do this is to create a project-level architecture for the system. This can be an ITS project architecture (as described later in the section on ITS architectures) or it could be a more detailed view of major components (meaning the subsystems or an additional level of decomposition) and communication links. At this step, the process identifies the ITS standards to support the interfaces that have been defined. (The topic of ITS standards is covered later in the module.)

Depending on the complexity of the system, requirements for each subsystem element may be defined and documented the same way as the system-level requirements. This step usually concludes with an important control gate review, sometimes called the preliminary design review.

Detailed Design

At the detailed design step (sometimes referred to as component design), the development team defines how the system will be built. At this step the focus is on *how* to define the system. To perform this step, each subsystem is further defined by components of hardware, software, database elements, firmware, and so on. For these components, detailed design specialists in the respective fields create documentation (build-to specifications) that will be used to build or procure the individual components. If custom hardware or software needs to be developed, specialists perform the detailed design of these components or software modules. In practice, most of the hardware used in ITS projects is existing vendor offerings, so little if any hardware design is required. Development of custom software is more common, but even here most ITS projects use vendor software that may be customized for the specific project. This detailed design includes the detailed definition of interfaces, which includes the definition of the communications to be used to link the various hardware subsystems or components in the project. Included in the interface definition and communications design is the detailed customization of the ITS standards to be used (see the later discussion of the ITS standards for more on this step). The control gate used for this step is sometimes called the critical design review. An additional output of this step is the creation of unit plans that will be used to verify that the units meet the detailed design specification.



Hardware and Software Procurement or Development

This stage involves hardware fabrication, software coding, database implementation, and the procurement and configuration of existing vendor products. The system's owner and stakeholders monitor this process with planned periodic reviews, for example, code



walkthroughs and technical review meetings. Concurrent with this effort, unit testing procedures are developed that will be used to demonstrate how the products meet the detailed design. At the completion of this stage, the developed products are ready for unit testing.

Unit Testing

The components from the hardware and software development are verified in accordance with the unit verification plan. The purpose of unit testing is to verify that the delivered components match the documented detailed design. In the case of the procurement of existing vendor hardware or software, unit testing would take the form of acceptance tests that are performed to show that the hardware and software meet the requirements that have been allocated to each unit.



Subsystem Integration and Verification

At this step, the components are integrated and verified at the subsystem level; that is, the subsystem meets its specified requirements. The first level of verification is done in accordance with the verification plan and is carried out in accordance with the verification procedures (step-by-step method for carrying out the verification) developed in this stage. Prior to the actual verification, a test readiness review may be held to determine that the subsystems are ready for verification. When it has been determined that verification can proceed, the subsystems are then verified. When the integration and verification are completed, the next level of subsystem is integrated and verified in the same manner. This process continues until all subsystems are integrated and verified.



One of the key features of subsystem integration and verification is its iterative nature. Normally a project will integrate a portion of the system, verify it through testing, then integrate some more of the system, verify it through testing, and so on until the complete system is put together. This process would be true for a complex system. However, projects with few components or subsystems will be able to quickly proceed through this step, illustrating again a key point of the SEP—that it needs to be tailored to each project; the process is not one size fits all.

System Integration and Verification

In systems engineering, there is a distinction between verification and validation. Verification confirms that a product meets its specified requirements. Validation confirms that the product fulfills its intended use.

In other words, verification ensures that you built the product right;

validation ensures that you built the right product. This is an important distinction, because there



are lots of examples of well-engineered products that met all of their requirements but ultimately failed to serve their intended purpose.

System verification is often done in two parts. The first part is done under a controlled environment (sometimes called a factory test). The second part is done within the environment in which the system is intended to operate (sometimes called onsite testing and verification) after initial system deployment. At this stage, the system is verified in accordance with the verification plan developed as part of the system-level requirements performed early in the development.

Initial System Deployment and System Validation

During deployment, the project development team installs the ITS in the operational environment and transfers its operation to the organization that will own and operate the system. The transfer includes support equipment, documentation, operator training, and other enabling products that support ongoing system operation and maintenance. Acceptance tests are performed as part of this step to confirm that the system performs as intended in the operational environment before control is transferred. This step may take several weeks to complete to ensure that the system operates satisfactorily in the long term. This is sometimes called a system burn-in. Many system issues surface when the system is operating in the real-world environment for an extended period of time.

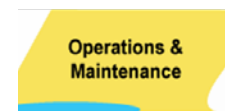
System Validation

After the ITS has passed system verification and is installed in the operational environment, validation tests are often run. Using the validation plan developed earlier, the system development team or the system owner or operator performs tests to determine if the deployed system meets the original needs identified in the concept of operations. The State DOT, a regional agency, or another entity also may choose to perform the tests. As a result of validation, new needs and requirements may be identified. This evaluation sets the stage for the next phase of the system.



Operations and Maintenance

After the initial deployment and system acceptance, the system moves into the Operations & Maintenance phase. In this phase the system will carry out the intended operations for which it was designed. During this phase, routine maintenance is performed as well as staff training. This phase is the longest phase, extending through the evolution of the system and ending when the system is retired or replaced. This phase may continue for decades. It is important that there are adequate resources to carry out the needed Operations & Maintenance activities; otherwise, the life of the system could be significantly shortened due to neglect.”⁹



Changes and Upgrades

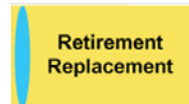
Once a system is in operation, it will still undergo various changes and upgrades. These changes and upgrades can be addressed by going back to the relevant steps of the V diagram and updating the outputs as required.



Using the V process for changes and upgrades will help maintain system integrity (synchronization between the system components and supporting documentation). When this approach is used, the systems engineering documentation produced in the original development can be updated to include the changes or upgrades. One of the outputs of the SEP is an accurate and complete set of documentation, which is extremely important to successful performance of this step. Depending on the nature of the changes and upgrades, the V process used may be an abbreviated one (e.g., no changes are needed to the ConOps, and only minor changes are needed to the requirements).

Retirement or Replacement

Eventually, every ITS will be retired or replaced for a variety of reasons. The system may no longer meet agency needs, or continuing its operation may not be cost-effective. Key system elements may be obsolete so it can no longer be maintained. Whatever the reason, the system owner will need to plan for its retirement or replacement. The next step may be to perform a study to assess the costs and benefits of retirement or replacement of the system. From a process viewpoint, this is equivalent to going back to the concept exploration step and revisiting the business case for the system. In a fitting symmetry to the V diagram, this final step is a planning step that can lead back to the beginning of a new project—a system replacement.



Cross-Cutting Activities

In addition to the steps previously covered, a number of cross-cutting activities are essential to a successful SEP. The term *cross-cutting* is used because these activities are not associated with a single step in the process but instead cut across many (or all) of the steps. The following are a few of the cross-cutting activities that occur when using systems engineering to develop almost any system.

Project Management

Project management is the process by which cost, scheduling, and resources are managed to successfully complete a project. It is a separate discipline that has its own defined process, along with a considerable body of knowledge and standards. The application of the project management process is essential to the successful completion of ITS projects. As such, project management and systems engineering work hand in hand during project development. Project management practices provide a supportive environment for the various development activities. The process provides the needed resources, then monitors and controls costs and schedules. It also communicates the project's status between and across the development team members, the system owner, and stakeholders.¹⁰ The system owner or agency has a vested interest in making sure that the development team (usually a contractor team) follows good project management practices, and they can require key outputs, such as the project management plan and project status reporting, as part of the contract deliverables.



Risk Management

Risk management is the identification and control of risks associated with the development effort. The goal of risk management is to identify potential problems before they occur, plan for their occurrence, and monitor the system development so that action can be taken early to prevent their occurrence. Identifying risks early is one of the key concepts of the SEP.



Risk management includes the following general steps:

- Risk identification. The objective of the risk identification step is to identify early on the key risks to project success. This will require the systems engineers, project management, and stakeholders to brainstorm where the risks may lie in any particular project. The key risk areas are technical aspects, cost, and schedule. In addition, risk identification should consider potential risks associated with the following:
 - Jurisdiction. Does the lead agency have jurisdiction or are there other responsible agencies?
 - Software. Is there existing and proven software that can be used?
 - Interfaces. Are new interfaces required or can a project rely on existing interfaces?
 - Requirements and procedures. Are these well defined and documented?
 - Experience. Does the staff have the necessary experience to procure, implement, and operate the project?
- Risk analysis and prioritization. Once risks have been identified, the next step in the process is to analyze and prioritize these risks. To do this, the developers need to answer three basic questions:
 - What will the impacts be if the risk does occur?
 - How severe will this impact be?
 - How likely is it that the risk will occur?
- Risk mitigation. For each risk identified, a plan needs to address the risk impacts should they occur.
- Risk monitoring. Finally a risk mitigation strategy is needed for monitoring risks so that the development team will be aware if the risks do in fact occur.

A risk management plan is usually developed at the beginning of the project (and is often included in the SEMP). However the plan should be reviewed and updated on a regular basis throughout the project to identify new risks and to assess whether the existing risks are occurring. One of the common mistakes made when executing projects is to create a risk management plan and not review and update it as the project proceeds.

Project Metrics

Project metrics are measures that are used to track and monitor the project and the expected technical performance of the systems development effort. The identification and monitoring of metrics allow the team to determine if the project is on track, both programmatically and technically. Programmatic metrics might be simple budget and schedule status. Technical metrics could involve measuring the number of requirements defined or the number of requirements changed over a period of time. These metrics are a key part of the project management process.



Configuration Management

Configuration management is defined as “a management process for establishing and maintaining consistency of a product’s performance, functional, and physical attributes with its requirements, design and operational information throughout its life.”¹¹ More than just defining the current state of a system, configuration management involves managing changes that are made to the system throughout its life.



USDOT provides the following definition: “Configuration management is the process that supports the establishment of system integrity (the documentation matches the functional and physical attributes of the system) and maintains this integrity throughout the life of the system (synchronizes changes to the system with its documentation).”¹²

Configuration management consists of several key aspects:

- Configuration management planning—defining the configuration management process to be used on the project. This plan is usually contained as a section of the SEMP.
- Configuration identification—identifying what items (i.e., documentation, hardware, and software) will constitute the “baseline” to be tracked as part of the official configuration of the system.
- Change management—defining the process used to incorporate changes to the baseline, including how and when changes are made.
- Status accounting—keeping track of the current status and any changes in process for each item of the baseline.

On any project, configuration management as a formal process commences when the first documentation is developed and continues as the project moves from one step to another in the SEP. Documentation is a part of the baseline and, once created, is subject to the change management process as defined.

Traceability

Traceability is a key concept of the SEP that ensures that the different outputs of the process properly relate to each other. Traceability centers on the requirements developed for the project and how they relate to other



outputs of the project. Each need in the ConOps must trace to a requirement in the system requirements document. If subsystem requirements are defined, then these must trace to system requirements. The requirements must also trace to verification cases. If design specifications are created, they need to trace back to the system requirements and forward (through test plans) to the verification that is done. As described above, traceability goes both backward (from requirements to needs) and forward (from requirements to specifications to verification). Traceability is the key concept in the SEP that ensures that the system that is implemented at the end of the project is directly connected with the user needs that were identified at the beginning of the project.

Related Topics

The following sections discuss a series of topics, including Federal regulations that are relevant to the use of the SEP in the development of ITS projects.

ITS Architecture

ITS architecture is a key component of the first step of the V process model. An ITS architecture is a framework for describing ITS-related transportation services. Three levels of ITS architecture are relevant to any discussion of the systems engineering process: the National ITS Architecture, regional ITS architectures, and ITS project architectures. Each of these types of architecture is discussed below.

National ITS Architecture

The National ITS Architecture provides a common framework for planning, defining, and integrating intelligent transportation systems. It is a mature product that reflects the contributions of a broad cross-section of the ITS community (transportation practitioners, systems engineers, system developers, technology specialists, consultants, and so forth).

The architecture defines the following elements:

- The functions (e.g., gathering traffic information or requesting a route) that are required for ITS
- The physical entities or subsystems where these functions reside (e.g., in the field or the vehicle)
- The information flows and data flows that connect these functions and physical subsystems into an integrated system

The National ITS Architecture is maintained and updated by USDOT in collaboration with ITS stakeholders. The first version of the National ITS Architecture was released in June 1996. The latest update, Version 7.0, was released by USDOT in January 2012

(www.iteris.com/itsarch/index.htm).

The National ITS Architecture is comprised of three architectural layers: Institutional, Transportation, and Communications:

The Institutional Layer includes the institutions, policies, funding mechanisms, and processes that are required for effective implementation, operation, and maintenance of an intelligent transportation system... The Transportation Layer is where the transportation solutions are defined in terms of the subsystems and interfaces and the underlying functionality and data definitions that are required for each transportation service. This layer is the heart of the National ITS Architecture. The Communications Layer provides for the accurate and timely exchange of information between systems to support the transportation solutions.

(Source: www.iteris.com/itsarch/html/archlayers/archlayers.htm.)

The layer of the National ITS Architecture most relevant to this discussion of systems engineering is the transportation layer, in which a logical (or function-based) architecture and a physical (or systems- and interconnections-based) architecture are defined. The top-level interconnect diagram of the physical architecture, which identifies the 22 subsystems of the architecture and the types of interconnects, is shown below. The physical architecture defines the information that flows between the key subsystems involved in delivering ITS services. These ITS services are described by a set of service packages that define the portion of the overall architecture required to provide a specific service (e.g., Traffic Incident Management). The overall pieces of the National ITS Architecture are more numerous and interconnected than described here. (For a more detailed view of the National ITS Architecture, refer to the website: www.iteris.com/itsarch/index.htm.)

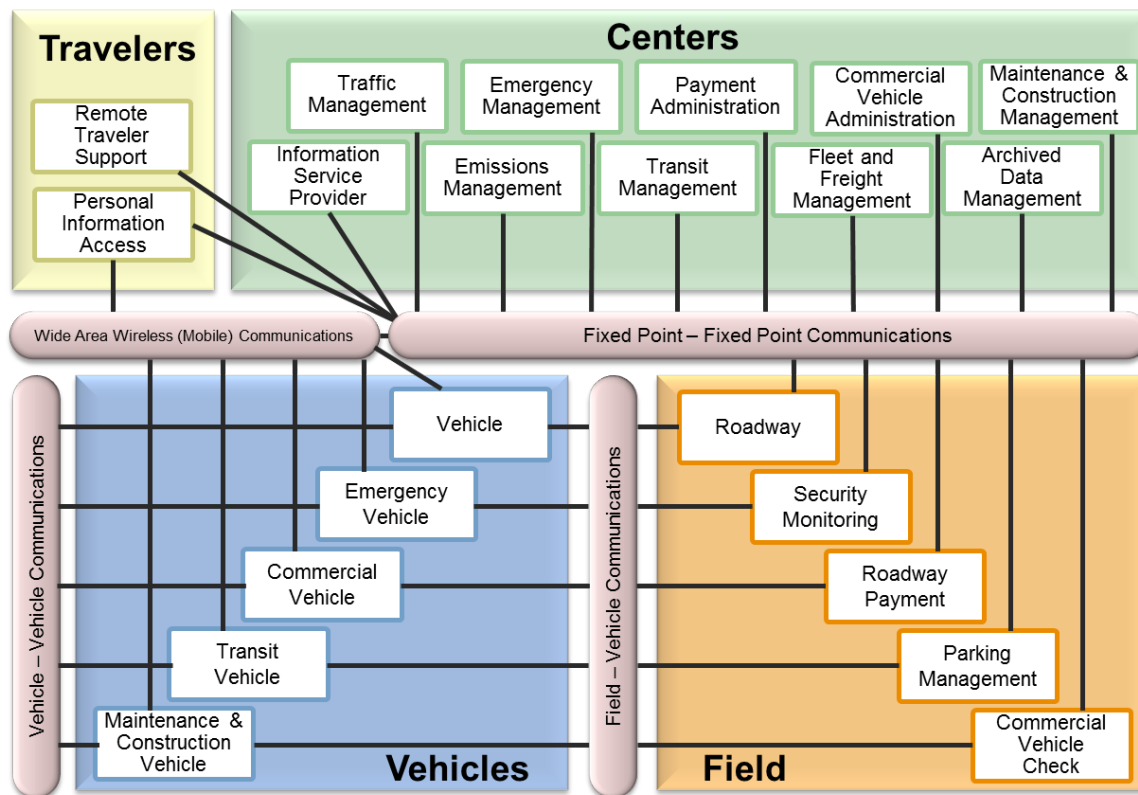
§ 940.9 Regional ITS Architecture.

The regional ITS architecture shall include, at a minimum, the following:

- (1) A description of the region;
- (2) Identification of participating agencies and other stakeholders;
- (3) An operational concept that identifies the roles and responsibilities of participating agencies and stakeholders in the operation and implementation of the systems included in the regional ITS architecture;
- (4) Any agreements (existing or new) required for operations, including at a minimum those affecting ITS project interoperability, utilization of ITS related standards, and the operation of the projects identified in the regional ITS architecture;
- (5) System functional requirements;
- (6) Interface requirements and information exchanges with planned and existing systems and subsystems (for example, subsystems and architecture flows as defined in the National ITS Architecture);
- (7) Identification of ITS standards supporting regional and national interoperability; and
- (8) The sequence of projects required for implementation.

Source: Rule 23 CFR 940.

Figure 4. National ITS Architecture



From the viewpoint of the SEP, the primary importance of the National ITS Architecture is the template that it provides in the form of subsystems, information flows, and service packages that form the basis of the regional ITS architectures that were discussed in the first step of the SEP. This template allows the creation of regional ITS architectures throughout the country that have a common underlying structure and definition.

Regional ITS Architecture

Whereas the National ITS Architecture guides ITS programs at the national level and addresses all subsystems, technologies, and standards, regional ITS architectures define the plans, programs, goals, and objectives for implementation on a more localized basis. Regional ITS architectures have been developed for States, metropolitan areas, or other regions of interest. They are developed through the participation of regional stakeholders, including owners and managers of highway and transit agencies, public safety agencies, motor carrier organizations, and other public transportation facilities.

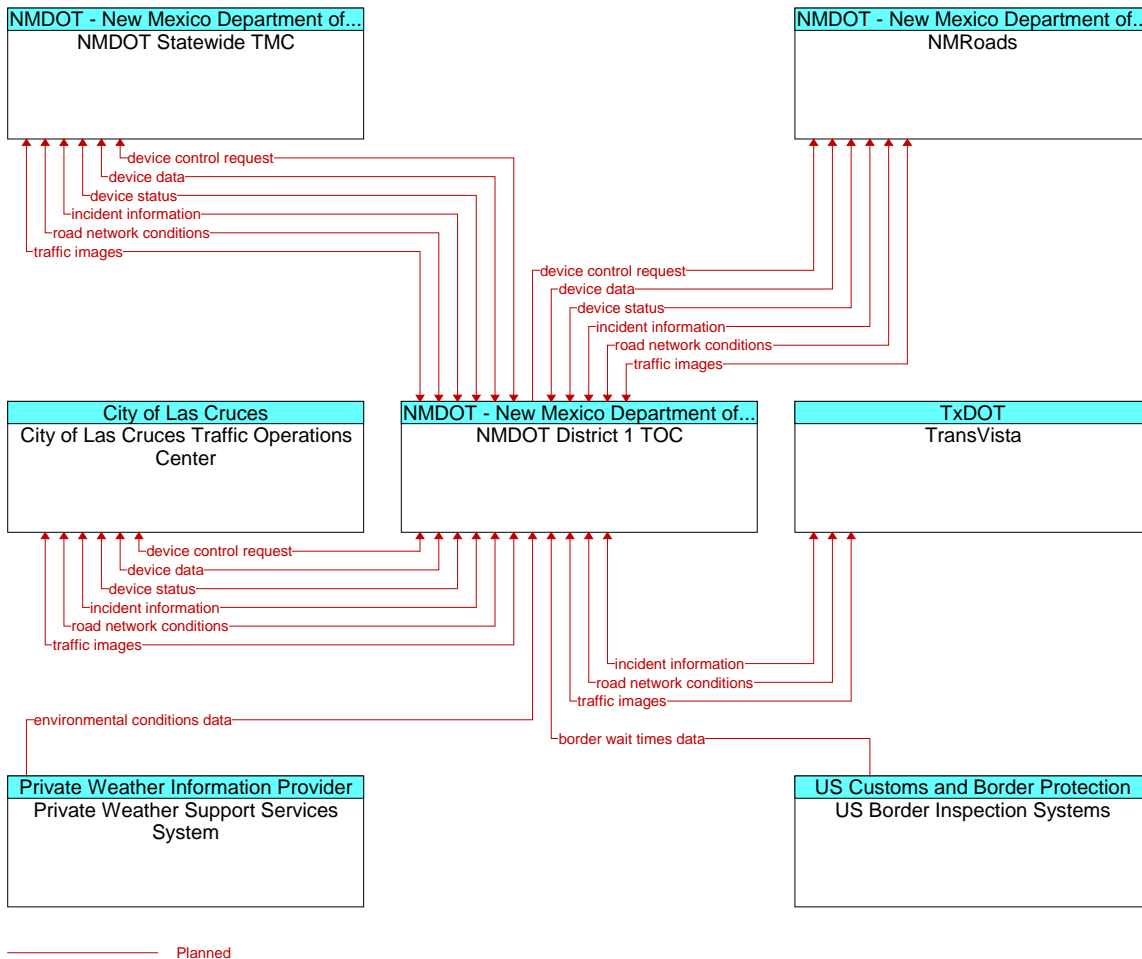
A regional ITS architecture is developed to meet the specific needs of a region; define program goals and stakeholder roles and responsibilities; and manage institutional agreements and technical integration of ITS within the region. It represents a regional adaptation of the National ITS Architecture. Creating a regional ITS architecture builds a shared vision for ITS implementation and advances regional transportation improvement programs and long-range transportation plans by planning operations and defining goals for regional ITS programs.

The components of a regional ITS architecture are defined by Rule 940 (and the corresponding FTA policy). FHWA has developed a regional ITS architecture development tool called Turbo Architecture, which is a database tool that holds the details of the region's stakeholders, systems, ITS services, system interconnects, projects, agreements, ITS standards, roles and responsibilities, and functional requirements. Additional information about regional ITS architectures, along with examples of each of the required outputs, can be found in the document *Regional ITS Architecture Guidance Document*.¹³

ITS Project Architectures

An ITS project architecture is a high-level view of a project that focuses on the systems, interfaces, and information flows relevant to the project. As discussed in the overview of the SEP, this is a typical output of the high-level design step. Although this type of architecture can be created to varying degrees of detail, one easy and effective way to create an ITS project architecture is to create the subset of the regional ITS architecture that relates to the project. Using this formulation, the project architecture would define stakeholders, systems, ITS services, interfaces, roles and responsibilities, agreements, and functional requirements relevant to the project. Many of these ITS project architecture outputs can be used to perform the systems engineering analysis required by another section of Rule 940 (see discussion below). Turbo Architecture can be a useful tool for creating the ITS project architecture. The following figure shows an example of a project architecture diagram created with Turbo Architecture from the New Mexico Statewide ITS Architecture. This project, for improvements to the District 1 Traffic Operations Center, shows the key interfaces that will be developed as part of the project.

Figure 5. Regional ITS Architecture



Source: Consensus Systems Technologies (ConSysTec).

Although ITS project architectures are usually produced from a regional ITS architecture, as illustrated above, in certain cases the project may not be described in the regional ITS architecture. In that case an ITS project architecture in a form similar to the regional ITS architecture can be produced and then used to help update the regional ITS architecture at its next iteration.

ITS Standards

The RITA ITS Standards website describes the standards as follows:

ITS standards define how ITS systems, products, and components can interconnect, exchange information and interact to deliver services within a transportation network. ITS standards are open-interface standards that establish communication rules for how ITS devices can perform, how they can connect, and how they can exchange data in order to interoperate. It is important to note that ITS standards are not design standards: They do not specify specific products or designs to use. Instead, the use of standards gives transportation agencies confidence that components from different manufacturers will work together, without removing the incentive for

designers and manufacturers to compete to provide products that are more efficient or offer more features. (www.standards.its.dot.gov/learn_WhatAre.asp)

ITS standards address the data transferred on an interface, the communications protocols used to send the data, and in a couple of cases the physical specification of hardware. From a data standpoint, the ITS standards focus on interfaces and information exchanges identified in the National ITS Architecture. Standards development is supported by the USDOT's Joint Program Office's (JPO) ITS Standards Program, which provides a collaborative process to define and update standards for use by all public and private entities in the development of ITS applications and technology. The ITS Standards Program works with standards development organizations such as the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), the American Public Transportation Association (APTA), the Institute of Electrical and Electronics Engineers (IEEE), and the National Electrical Manufacturers Association (NEMA) to address interface requirements between different ITS applications.

ITS standards provide the technical guidance and requirements for each component of an ITS system. They guide every aspect of technical applications and system communications, and compliance is required for all applications. To properly specify ITS standards, officials must customize them to fit the needs and requirements of the project. More information about ITS standards and how to specify them for projects is available on the USDOT Professional Capacity-Building website. A set of ITS Standards Training modules can be found at www.pcb.its.dot.gov/stds_training.aspx.



Using the SEP for ITS Project Development

The SEP, as discussed earlier, may be viewed as an extension of the traditional project development process that is already established in transportation agencies. Although project development processes vary from State to State and from organization to organization within each State, the transportation project development process tends to have the same major steps.

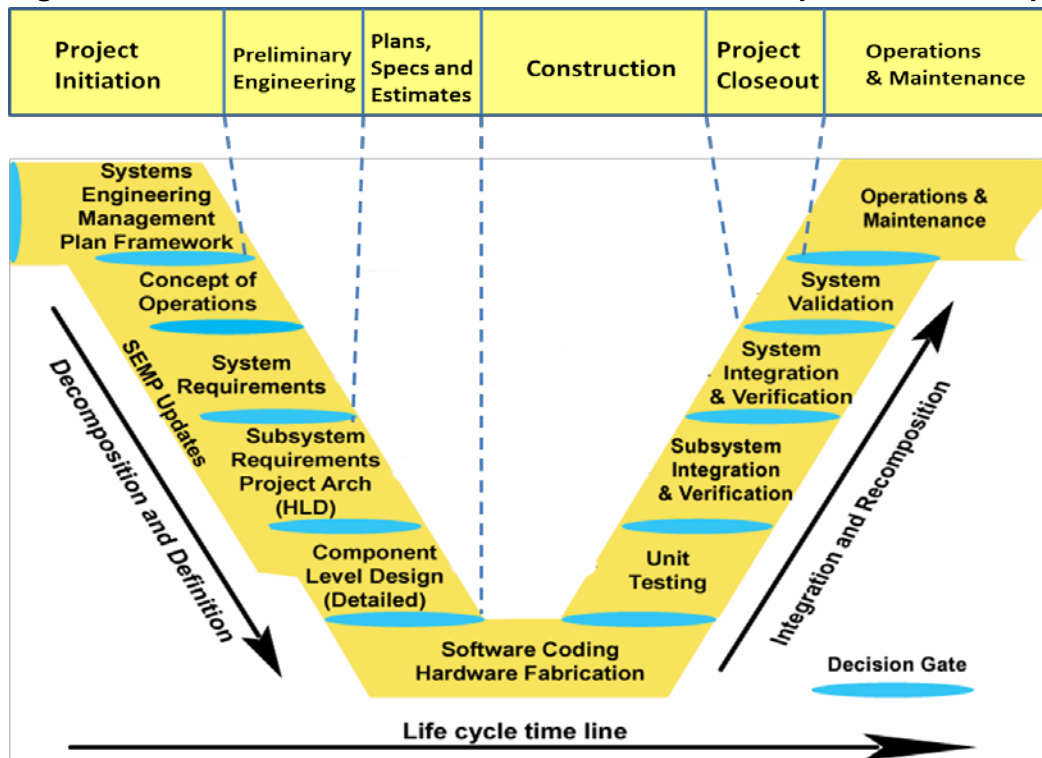
- **Project Initiation.** In this step, the project manager is identified, the project team is assembled, and the project development is planned. A high-level scope of the project is developed, costs are estimated, and the required forms and checklists are completed to garner approval for the project from the sponsoring and funding agency(ies). For FHWA and FTA, this is a critical point in the process where approval to proceed is given and Federal funds are obligated.
- **Preliminary Engineering.** In the traditional capital project development process, environmental, right-of-way, and other studies are performed depending on the type of project. These studies result in better understanding of the project requirements and constraints. ITS projects that include a construction component will require these same

studies as well as additional engineering analyses to fully specify the project requirements for the ITS portion of the project.

- Plans, Specifications, and Estimates. The detailed design for the project, complete with detailed project specifications, estimates of material needs, and associated costs, are documented. In a traditional construction project, this process step provides companies with all the information they need to develop an accurate bid.
- Construction. The project is built. For a traditional transportation project, this is construction of the actual physical improvement. For an ITS project, this includes the procurement and implementation of the actual hardware, software, and enabling products (e.g., manuals, operating procedures, and training).
- Project closeout. After final inspection and testing, the completed project is accepted, as-built plans are created, and a project history file is completed.

There is a close correspondence between the SEP and the traditional transportation development process as shown in Figure 6.

Figure 6. Correlation Between SEP and Traditional Transportation Development Process



The project initiation phase includes activities that occur prior to the start of the project, which includes the procurement process. The impact of the SEP on ITS project procurement is described in the document Model Systems Engineering Documents for Adaptive Signal Control Technology (ASCT) Systems.¹⁴ The preliminary engineering corresponds to the concept of operations and system requirements steps. The step of plans, specifications, and estimates relates to the design steps. Finally the construction step is analogous to the development and verification steps of the SEP.

Systems Engineering Analysis Requirements

Rule 940 (Section 940.3 Definitions) contains specific requirements related to performing systems engineering analysis (SEA) for ITS projects that use highway trust funds (see sidebar). The rule defines an ITS project 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.”

Several key observations can be made about these SEA requirements. For example, item (b) says that the analysis should be on a scale commensurate with the scope of the project. This is a key concept in the SEP—that the level of effort should be tailored to the scope of the project. As has been mentioned before in this module, the SEP is not a one-size-fits-all process; rather, the effort expended should be related to the scope of the project. One important way to assess the level of effort required is to consider the risk associated with the project. A very low-risk project (e.g., installation of additional closed-circuit television cameras to an existing system) might proceed with a minimum SEP, possibly referencing work done in previous stages of the project. However, a project with more inherent risk (e.g., the initial development of a Traffic Operations Center in a region) would likely want to go through the process step by step.

The requirements in Section 940.11 correspond to the SEP, but not identically, because Rule 940 came out in 2001, prior to the development of the SEP for transportation projects that is referenced in this module. However, if the SEP steps are followed as described above, all of the requirements of the rule will be addressed by the steps from concept of operations through detailed design. Many States have developed Systems Engineering Review Forms based on the requirements, which can be used as a checklist to identify how systems engineering outputs on a particular ITS project address the Rule 940 SEA requirements.

Relationship to Transportation Planning

Although the majority of the steps in the SEP address the life cycle of a specific project (moving from systems engineering management planning to system validation), the wings of the V diagram affect the planning of ITS systems. This planning occurs within the structure of the overall transportation planning process.

Transportation planning is generally performed by State DOTs and metropolitan planning organizations (MPOs). One key output of the planning process is a long-range transportation plan (LRTP). This plan is sometimes referred to as a metropolitan transportation plan (MTP) at the MPO level and a long-range statewide transportation plan at the State level. In metropolitan areas, the transportation plan is the statement of the ways the region plans to invest in the transportation system. According to the Federal regulation, the plan shall “include both long-range and short-range program strategies/actions that lead to the development of an integrated intermodal transportation system that facilitates the efficient movement of people and goods.” The second key output of the planning process is the Transportation Improvement Program (TIP). At the two levels discussed in this module, these are usually called the Metropolitan TIP or Statewide TIP. In the TIP, the MPO identifies the transportation projects and strategies from

the MTP that it plans to undertake over the next four years. All projects receiving Federal funding must be in the TIP. The TIP is the region's way of allocating its limited transportation resources among the various capital and operating needs of the area, based on a clear set of short-term transportation priorities.¹⁵

The regional ITS architecture discussed in the first step of the SEP represents a regional plan for the deployment of ITS. The architecture represents the wish list for ITS because it contains a list of ITS projects without fiscal constraint. Because of the timeframe of the regional ITS architecture, it will usually contain outputs that can be used in the development of both the LRTP and the TIP. For example, the architecture usually contains a list of medium- and long-term projects, which represent initiatives that have not yet been funded, and as such they can be an input to the development of the TIP. If the projects are described in the more general terms of strategies, they can provide an input to the strategies described in the LRTP. Because ITS systems have life cycles that are often less than 10 years, transportation planning needs to include explicit consideration of technology updating or replacement, for example, as system preservation projects in the regional ITS architecture. A more complete discussion of how the regional ITS architecture supports transportation planning can be found in the *Regional ITS Architecture Guidance Document*¹⁶ and in *Applying a Regional ITS Architecture to Support Planning for Operations: A Primer*.¹⁷

Both the concept exploration and the retirement or replacement steps of the SEP provide outputs that support ITS planning. In the concept exploration step, the output of the effort will define the technical and cost parameters for projects or initiatives to be included in the TIP or LRTP. As described earlier, this step looks at alternative concepts to find the one with the best cost-to-benefit ratio or business case. In the retirement or replacement step a study or analysis is performed to determine the best approach to addressing a particular system's end of life. The results of the analysis or study are then moved to either the LRTP (if the resulting replacement is further in the future) or to the TIP (if the replacement is ready to be programmed as a TIP project).

Summary

This module has provided a step-by-step overview of the systems engineering process as it is applied to the ITS field. Related topics of the National ITS Architecture and ITS Standards Program were addressed along with their relationship to the SEP. Finally, the module described the relationship of the SEP to ITS project development and ITS planning.

References

1. Project Apollo: A Retrospective Analysis, <http://history.nasa.gov/Apollo/Apollo.html>
2. *INCOSE Systems Engineering Handbook*, Version 3.1, 2007.
3. Federal Highway Administration (FHWA) Final Rule 940 on ITS Architecture and Standards Conformity, and the Federal Transit Administration (FTA) Policy on ITS Architecture and Standards Conformity, enacted January 8, 2001, www.ops.fhwa.dot.gov/its_arch_imp/policy.htm
4. *Systems Engineering for ITS—An Introduction for Transportation Professionals*, USDOT, September 2007, <http://ops.fhwa.dot.gov/publications/seitsguide/seguide.pdf>

5. *Systems Engineering Guidebook for ITS*, Version 3.0, FHWA and Caltrans, November 2009, www.fhwa.dot.gov/cadiv/segb/
6. ANSI/AIAA G-043A-2012, *Guide for the Preparation of Operational Concept Documents*, American Institute of Aeronautics and Astronautics, 2012.
7. ANSI/GEIA EIA-632, *Processes for Engineering a System*, Sept 1, 2003.
8. *Systems Engineering for ITS—An Introduction for Transportation Professionals*.
9. *Systems Engineering Guidebook for ITS*, Version 3.0.
10. *A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide)*, 2013.
11. ANSI/EIA 649-1998, National Consensus Standard for Configuration Management.
12. *Systems Engineering for ITS—An Introduction for Transportation Professionals*.
13. *Regional ITS Architecture Guidance Document*, Version 2.0, USDOT, July 2006, www.ops.fhwa.dot.gov/publications/regitsarchguide/index.htm
14. Model Systems Engineering Documents for Adaptive Signal Control Technology (ASCT) Systems, FHWA, FHWA-HOP-11-027, August 2012, <http://ops.fhwa.dot.gov/publications/fhwahop11027/>
15. *The Transportation Planning Process, Key Issues: A Briefing Book for Transportation Decisionmakers, Officials, and Staff*, USDOT, 2007, www.planning.dot.gov/documents/briefingbook/bbook.htm
16. *Regional ITS Architecture Guidance Document*.
17. *Applying a Regional ITS Architecture to Support Planning for Operations: A Primer*, USDOT, FHWA-HOP-12-001, February 2012, www.ops.fhwa.dot.gov/publications/fhwahop12001/

Resources

1. IEEE 1028-1997, IEEE Standard for Software Reviews, Institute of Electrical and Electronics Engineers, 1998.
2. IEEE 1220-2005, IEEE Standard for Application and Management of the Systems Engineering Process, Institute of Electrical and Electronics Engineers, Sept 9, 2005.
3. IEEE Std 1362-1998, IEEE Guide for Information Technology – System – Definition – Concept of Operations (ConOps) Document, Institute of Electrical and Electronics Engineers, 1998.
4. ISO/IEC 15288:2008, *Systems and Software Engineering—System Life Cycle Processes*.