Deliverable 1-10: Technical Memorandum

Florida's Statewide Systems Engineering Management Plan

March 7, 2005
Version 2

Prepared for:

Florida Department of Transportation
Traffic Engineering and Operations Office
Intelligent Transportation Systems (ITS) Section
605 Suwannee Street, M.S. 90
Tallahassee, Florida 32399-0450
(850) 410-5600

This page is intentionally left blank.

DOCUMENT CONTROL PANEL		
File Name:	*Deliverable 1-10: Technical Memorandum – Florida's Statewide Systems Engineering Management Plan*	
File Location:		
Deliverable Number:	1-10	
Version Number:	2	
	Name	**Date**
Created By:	John M. Bonds, PBS&J	December 11, 2003
	Troy Nguyen	December 11, 2003
	Robert A. Nielsen	February 12, 2004
	Martin S. Sas, PBS&J	March 24, 2004
	Christopher S. Bausher, PBS&J	April 18, 2004
Reviewed By:		
Modified By:	Dave Hodges, PBS&J	
	Pam Hoke, PBS&J	June 9, 2004
	John M. Bonds, PBS&J	November 16, 2004
	Pam Hoke, PBS&J	February 28, 2005
Completed By:	Pam Hoke, PBS&J	March 7, 2005

This page is intentionally left blank.

Table of Contents

CHAPTER 1 – PROGRAM OVERVIEW

CHAPTER 2 – REFERENCED DOCUMENTS

CHAPTER 3 – THE SYSTEMS ENGINEERING PROCESS DEFINED

CHAPTER 4 – MANAGEMENT WITH THE SYSTEMS ENGINEERING PROCESS

CHAPTER 5 – CONCEPT OF OPERATIONS DOCUMENTATION

CHAPTER 6 – ENGINEERING SPECIALTY INTEGRATION

CHAPTER 7 – FLORIDA'S STATEWIDE SEMP SUPPORT PLANS AND PRACTICES

CHAPTER 8 – TAILORING AND SCALABILITY GUIDELINES

List of Appendices

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

List of Figures

List of Acronyms

ANSI .... American National Standards Institute
ARO .... Assurance Review Office
ATIS .... Advanced Traveler Information System
ATM .... Asynchronous Transfer Mode
ATMS .... Advanced Traffic Management System
AVL .... Automated Vehicle Location
BAFO .... Best and Final Offer
BIT .... Built-In Testing
C2C .... Center-to-Center
CCTV .... Closed-Circuit Television
CD .... Compact Disk
CDMP .... Configuration and Data Management Plan
CDR .... Critical Design Review
CDRL .... Contract Deliverable Requirements List
CD-ROM .... Compact Disk – Read-Only Memory
CFP .... Cost Feasible Plan
CFR .... *Code of Federal Regulations*
CHDR .... Critical Hardware Design Review
CI .... Configuration Item
CM .... Configuration Management
CMAQ .... Congestion Mitigation and Air Quality
CMB .... Change Management Board
CMP .... Configuration Management Plan
CMS .... Congestion Management System
ConOps .... Concept of Operations
COOR .... Concept of Operations Review
CORBA .... Common Object Request Broker Architecture
COSHH .... Control of Substances Hazardous to Health
COTS .... Commercial Off-the-Shelf
CRL .... Critical Resource List
CSA .... Configuration Status Accounting

CSCI.. Computer Software Configuration Item
CSDR ....Critical Software Design Review
CSO ....Contractual Services Office
CWBS ....Contract Work Breakdown Structure
D/I ....Drop / Insert
DDE ....Data Dictionary Entry
DFD...Data Flow Diagram
DID ....Data Item Description
DMS...Dynamic Message Sign
DOS.. Denial of Service
DPP ....Detailed Project Plan
ECO... Engineering Change Order
ECP ....Engineering Change Proposal
ECR...Engineering Change Request
EDL...Electronic Documents Library
EEPROM .... Electrically Erasable Programmable Read-Only Memory
EIA...Electronic Industries Alliance
EMC...Electromagnetic Compatibility
EMC..Emergency Management Center
EMI ....Electromagnetic Interference
EPROM..Erasable Programmable Read-Only Memory
ETA... Estimated Time of Arrival
EV ....Emergency Vehicle
FAC...Florida Administrative Code
FAT ....Final Acceptance Test
FCA..Functional Configuration Audit
FDOT .... Florida Department of Transportation
FDR...Final Design Review
FHDR..Final Hardware Design Review
FHWA ....Federal Highway Administration
FIHS...Florida Interstate Highway System
FMEA ....Failure Mode Effective Analysis
FMECA.. Failure Mode, Effect, and Criticality Analysis
FRACAS ....Failure Reporting, Analysis, and Corrective Action System

FS .... *Florida Statutes*
FSDR .... Final Software Design Review
FTA .... Federal Transit Administration
FTP .... File Transfer Protocol
GC .... General Consultant
GDT .... Geographic Data Technology
GEC .... General Engineering Consultant
GIS .... Geographic Information Systems
GPS .... Global Positioning System
GUI .... Graphical User Interface
HCI .... Human-Computer Interface
HDD .... Hardware Design Document
HDP .... Hardware Development Plan
HDR .... Hardware Design Review
HFE .... Human Factors Engineering
HFEPP .... Human Factors Engineering Project Plan
HIPO .... Hierarchal Input-Process-Output
HOV .... High Occupancy Vehicle
HRR .... Hardware Requirements Review
HWCI .... Hardware Configuration Item
I&T .... Integration and Testing
I-95 .... Interstate 95
IC .... Integration Case
ICD .... Interface Control Document
ICS .... Interface Control Specifications
IEC .... International Electrotechnical Commission
IEEE .... Institute of Electrical and Electronics Engineers
ILS .... Integrated Logistics Support
ILSP .... Integrated Logistics Support Plan
IMMS .... Incident Management Message Sets
IMS .... Integrated Master Schedule
INCOSE .... International Council on Systems Engineering
IP .... Internet Protocol
IPT .... Integrated Product Team

IS .... Interim Standard
ISO .... International Organization for Standardization
IT .... Information Technology
ITN .... Invitation to Negotiate
ITS .... Intelligent Transportation System
IV&V .... Independent Verification and Validation
LED .... Light-Emitting Diode
LRTP .... Long-Range Transportation Plan
LRU .... Lowest Replaceable Unit
LSA .... Logistics Support Analysis
MACA .... Months After Contract Award
MDX .... Miami-Dade Expressway
MIL-STD .... Military Standard
MMI .... Man-Machine Interface
MOE .... Measure of Effectiveness
MOP .... Measure of Performance
MOU .... Memorandum of Understanding
MPO .... Metropolitan Planning Organization
MTBF .... Mean Time Between Failures
MTTR .... Mean Time to Repair
NHS .... National Highway System
NIAP .... National Information Assurance Partnership
NIST .... National Institute of Standards and Technology
NITSA .... *National Intelligent Transportation Systems Architecture*
NSA .... National Security Agency
O/HE .... Operability/Human Engineering
ODP .... Operational Development Plan
OER .... Octet Encoding Rules
ORR .... Operational Readiness Review
PCA .... Physical Configuration Audit
PCE .... PERT Chart Expert®
PDR .... Preliminary Design Review
PERT .... Project Evaluation and Review Technique
PHDR .... Preliminary Hardware Design Review

PIR .... Process Improvement Review
PMP .... Program Management Plan
PMPT .... Project Master Plan Task
POS .... Point of Sale
PPMP .... Project Performance Management Plan
PPMR .... Program Performance Management Report
PRISMA .... Program Review for Information Security Management Assistance
PROM .... Programmable Read-Only Memory
PS&E .... Plans, Specifications, and Estimates
PSDR .... Preliminary Software Design Review
PSpec .... Process Specification
PTR .... Problem Tracking Report
PTZ .... Pan-Tilt-Zoon
QA .... Quality Assurance
QC .... Quality Control
QCAP .... Quality Control Assurance Process
QM .... Quality Management
QMP .... Quality Management Plan
R&M .... Reliability and Maintainability
RF .... Radio Frequency
RFI .... Request for Information
RFP .... Request for Proposal
RFQ .... Request for Qualifications
RITSA .... Regional Intelligent Transportation Systems Architecture
RMP .... Risk Management Plan
RMPP .... Reliability and Maintainability Program Plan
ROM .... Rough Order of Magnitude
RTM .... Requirements Traceability Matrix
RTMC .... Regional Transportation Management Center
RTVM .... Requirements Traceability Verification Matrix
SANS .... SysAdmin, Audit, Network, and Security
SAT .... System Acceptance Test
SBS .... System Breakdown Structure
SCM .... Software Configuration Management

SDD....................System Design Document
SDP .... Software Development Plan
SDR....................Systems Design Review
SDRL .... Subcontractor Data Requirements List
SECM.................... Systems Engineering Capability Model
SEIT ....Systems Engineering Integration Team
SEMP ....Systems Engineering Management Plan
SEP.................... Systems Engineering Process
SIC ....SunGuide^SM ITS Checklist
SIT....................Systems Integration and Testing
SITSA....................*Statewide Intelligent Transportation Systems Architecture*
SME ....Subject-Matter Expert
SMP.................... Subcontract Management Plan
SOW....................Statement of Work
SQA.................... Software Quality Assurance
SRR .... System Requirements Review
SSDD .... Software System Design Document
SSDR....................Software System Design Review
SSR .... System Requirements Review
STIP .... State Transportation Improvement Plan
STP....................System Test Plan
STP/U....................Surface Transportation Program/Urban
SUT ....System Under Test
SWDR....................Software Design Review
SWRR ....Software Requirements Review
T&M .... Time and Materials
TC ....Test Case
TCP .... Transmission Control Protocol
TEA-21 .... *Transportation Equity Act for the 21st Century*
TEOO.................... Traffic Engineering and Operations Office
TERL.................... Traffic Engineering Research Laboratory
TIP....................Transportation Improvement Program
TLC .... Through-Life Cost
TMC.................... Transportation Management Center

Appendix E

Program Management Plan Template

This page is intentionally left blank.

Title Page
Document Control Panel
Table of Contents
List of Acronyms
Definitions

1. Introduction

The Program Management Plan (PMP) establishes the management approach used on an FDOT ITS project that is consistent with the approach used on all FDOT ITS programs. The PMP describes the overall program structure; deliverables; related management plans and procedures; and the methods used to plan, monitor, control, and improve the project development efforts. The PMP is a dynamic document and is updated on a periodic basis to reflect all organizational changes, lessons learned, and advances in methodologies that occur throughout a project's life cycle.

1.1 Project Information

This section provides a brief summary of the program and includes:

- Project Name
- Contract Number
- Period of Performance
- Contract Type (i.e., firm fixed price, cost plus, etc.)
- Contract Value
- Delivery Date
- Project Manager
- Contact Information

1.2 Deliverables

1.2.1 System Description

This section provides a brief description of the system being developed and managed.

1.2.2 Products and Services

This section describes what products are being supplied to the FDOT under the contract. It also provides a list of products and services that will be produced by an ITS project.

1.3 Key Project Events

This section describes the key milestone events for an ITS project. For example, functional assessment events, acceptance/approval events, key technical demonstrations/collaborative events, etc.

1.4 Goals

This section briefly describes any strategic and/or tactical goals a project is to achieve. These may include cost, schedule, or technical goals.

1.5 Vision and Charter

This section briefly describes the project's vision and charter. The purpose of a shared vision is to provide a statement of an envisioned future, and to establish a common understanding of the aspirations and governing ideals of the project. The charter is the contract among the project members for the expected work effort and level of performance.

2. Organization Structure

This section describes the organizational structure of the project. The organization includes FDOT personnel, subcontractors, customers, and users. The description includes the role and involvement of each entity in the project.

2.1 Responsibilities

This section describes the roles and responsibilities of project personnel, including the:

- Project Manager
- Subcontracts Manager
- Project Engineer
- Finance Manager
- Planning
- Operation
- Procurement
- Configuration Management
- Quality Assurance

3. Management Plans

3.1 Statement of Work

The statement of work (SOW) defines the planning and management activities that will be expected by the customer. The SOW is mutually agreed to by the customer and FDOT project management. The SOW resides in the project library and is maintained by the project manager

3.2 Cost Management

This section describes internal cost management policies that provide a set of operating procedures for planning, directing, monitoring, and measuring work. These policies provide controls for accurate decision-making data for the project management team to analyze, capture variances, and plan revisions to the baseline costs.

3.3 Schedule Management

The integrated master schedule outlines the program plan in sufficient detail to define resource requirements, material timing, and integration requirements with existing plans and schedules. The integrated master schedule, built around the contract work breakdown structure (CWBS), is the top level of the scheduling system, and is supported by a hierarchy of intermediate and detailed schedules. All critical dependencies, resources needed, and critical path items are clearly identified on the integrated master schedule.

3.4 Technical Management

The technical management strategy for the program is documented in three key management plans, including the:

- **Systems Engineering Management Plan (SEMP) –** This plan describes the overall plans for the engineering and manufacturing development of each program. The SEMP should also describe how the technical baselines (e.g., requirement specifications, etc.) for the program will be documented, traced to other engineering work products, and maintained.

- **Software Development Plan (SDP) –** The SDP establishes the software development approach, methodologies, tools, and procedures to be used during the analysis, design, development, testing, integration, deployment, and maintenance of the software for each FDOT project.

- **Hardware Development Plan (HDP) –** The HDP establishes the hardware development approach, methodologies, tools, and procedures to be used during the analysis, design, fabrication, testing, integration, deployment, and maintenance of the hardware for each FDOT program.

The SEMP, SDP, and HDP serve as the program's implementation of the tailored FDOT standard organizational SEP. These documents reside in the project library and are maintained by the project engineer.

3.5 Resource Management

Resource management (i.e., staffing) on the project is handled by the FDOT project manager and subcontract manager, as applicable. The FDOT ITS project staffing needs will be reviewed on a monthly basis and subcontractor staffing will be statused through monthly progress reviews conducted by the subcontract manager.

3.6 Risk Management

Risk management is designed to reduce the impact of programmatic and technical uncertainties to acceptable and manageable levels. Risks are inherent in every endeavor, especially during the development phases of a program's life cycle. Each person on the program is responsible for overall risk management, and uses the same process to identify and control risks.

3.7 Subcontract Management

If the project utilizes subcontracted services and products, it is essential to have an experienced subcontract manager, subcontractor SOW, and a subcontract management plan. The objective of these documents and personnel is to ensure purchase orders are properly executed, providing the appropriate controls to meet the program schedule, budgets, and technically compliant services and products to review the progress of the subcontractor and subcontract management activities.

3.8 Configuration and Data Management

The configuration and data management activities for the program are documented in the configuration data management plan (CDMP). The CDMP describes the procedures to be used on the program to assure the integrity and control of the products being developed (e.g., configuration identification and methods; formal release configuration data management controls; development of library controls; the engineering change proposal (ECP) process; change control process; program problem reporting; configuration control board establishment; configuration audits; and the storage, handling, and delivery of project media).

3.9 Quality Management

The quality management (QM) activities for the program are documented in the quality management plan (QMP). The QMP describes the procedures to be used on the program to implement a quality program and provide FDOT management personnel with visibility into the quality of the products (e.g., process and product evaluations; record keeping; nonconformance tracking; and reporting channels).

3.10 Monitoring and Control

Monitoring and control activities are used to monitor the project's performance. These include the uses of periodic status reviews; informal and formal reviews; milestone reviews; and metrics analysis to determine the health of the program.

4. Statusing

This section describes the progress reviews, scheduled as periodic occurrences or on an as-needed basis, used on the project to status the program's performance. These reviews include the results and analyses of reviewing internal and external commitments, risks, management of project data, and stakeholder involvement. These reviews are performed at various project levels and may include both project and senior management representation.

Results, documented as meeting minutes, include a minimum of the decisions made during the meeting, attendance, issues, risks, and action items.

4.1 Reviews

This section describes the typical mechanisms employed on the project that are initiated from either internal or external needs.

- **Initial Baseline Reviews (IBRs)** are used to assess the project's technical, cost, and schedule baselines. Initial baseline reviews provide an independent mechanism to assure project management that the project has the infrastructure in place to meet its performance commitments for the project to remain on schedule and under budget. An example of an IBR is the project kickoff meeting where the project plan is reviewed.

- **Independent Technical Assessments (ITAs)** are used to assess proposed technical, cost, schedule, and staffing objectives for new business opportunities or enhancements. Independent technical assessments are used to identify risks prior to the establishment of a contractual commitment. These assessments provide an independent review mechanism to assure project management that the proposed commitment is complete, meets FDOT policy, and is well thought out. An ITA may be conducted by an FDOT contractor that has no relationship to the development contractor or by a separate FDOT committee.

- **Performance Assessment Reviews (PARs)** are used to assess the project technical, cost, and schedule performance against the baseline plans. Performance assessment reviews provide an independent mechanism to assure project management that the project performance commitments are met in order for the project to maintain a "GREEN" status. Performance assessment reviews help identify weaknesses in the project's implementation of the baseline plans and help foster an atmosphere of continuous improvement. Examples of PARs are the monthly progress reviews; the system requirements review; the system design review; software and hardware design reviews; etc.

- **Milestone Reviews** are typically formal reviews. Reviews are conducted at meaningful points in the project's schedule, such as the completion of selected stages (or phases), with relevant stakeholders (i.e., managers, staff members, customers, end users, suppliers etc.). Project commitments, plans, status, and risks are reviewed. Significant issues and their impact are identified and documented. Results of the review, such as action items and decisions, are documented as review minutes. Action items are tracked to closure. Examples of milestone reviews include the system requirements review (SRR), system design review (SDR), software specification review (SSR), preliminary design review (PDR), critical design review (CDR), test readiness review (TRR), etc.

4.2 Metrics

This section describes the metrics used to provide adequate quantitative visibility into the development progress so that project management can take effective actions when a project's performance deviates from its plans. Actual performance metrics are tracked against documented estimates, commitments, and plans, and the appropriate action is taken when plans and actuals deviate.

The project has identified a set of project metrics to be used for project progress management, project quantitative management, and the associated analysis activities (e.g., sourcing, delivery performance, and other project metrics). These metrics, along with the project's quantitative objectives, are documented in the project's program performance management plan (PPMP). Refer to *Appendix O* for a template to use in creating this plan.

In addition, the project produces a monthly program performance management report (PPMR) that describes the quantitative analysis performed on selected processes in accordance with defined project goals. Together with other FDOT projects, these metrics are used in the planning and costing of future activities as well as to support ongoing process evaluations.

This page is intentionally left blank.

TPM .... Technical Performance Measure
TRR.. Test Readiness Review
TWO .... Task Work Order
TxDOT .... Texas Department of Transportation
USDoD.. United States Department of Defense
USDOT .... United States Department of Transportation
UUT .... Unit Under Test
VE .... Value Engineering
VIDS .... Vehicle Image Detection System
WBS .... Work Breakdown Structure

This page is intentionally left blank.

CHAPTER 1 – PROGRAM OVERVIEW

1. Introduction

The establishment of the Florida Department of Transportation (FDOT) Intelligent Transportation Systems (ITS) Section marked the beginning of FDOT's public commitment to increase the efficiency of its transportation systems using innovative new approaches being developed and tested around the country. These new approaches concentrate on using electronic devices, computer networks, and communication systems to collect and process information related to roadway and traffic conditions. This information is then relayed back to motorists, allowing them to monitor conditions and make informed decisions regarding their routes. The systems that collect, process, and relay the information are collectively called intelligent transportation systems (ITS).

Implementing a standard engineering process for the deployment of ITS projects in Florida will increase the likelihood of a project's successful deployment. Studies have shown that the overall success rate for projects without some form of management plan, such as systems engineering, is just over 15 percent, while the remainder of the projects were either cancelled or deemed inadequate.[1] The underlying causes for these projects' inadequacies or cancellations were related to deficiencies in the management of quality, schedule, and budget. Because those project elements are the primary concern of systems engineering, the implementation of *Florida's Statewide Systems Engineering Management Plan* (*SEMP*) is expected to improve the success rate of ITS deployments.

1.1 Why Read and Use Florida's Statewide SEMP?

Florida's Statewide SEMP is both a reference tool and a training manual that will help District ITS engineers and ITS project administrators comply with federal regulations for federal-aid ITS projects. By reading the *SEMP*, you will understand the difference between a Federal Highway Administration (FHWA) project-level oversight "nonexempt" ITS project and a FHWA program-level oversight "exempt" ITS project.

Florida's Statewide SEMP will explain the roles and responsibilities of the ITS project manager or administrator and, most importantly, will specify the steps involved in the implementation and management of a successful ITS project.

---

[1] Federal Highway Administration NHI-02-025, *Introduction to Systems Engineering for Advanced Transportation*, Course No. 137024.

The *SEMP* will explain the ITS project approval process for Florida based on the federal and state requirements.

1.2 Compliance with Title 23, Part 940, of the Code of Federal Regulations

Section 5206(e) of the *Transportation Equity Act of the 21st Century* (*TEA-21*)[2] was enacted June 9, 1998, and requires ITS projects that are funded through the Highway Trust Fund to conform to the *National ITS Architecture* (*NITSA*).[3] It further requires the *NITSA* to be used to develop a local implementation plan called a regional architecture. The date *TEA-21* became effective was February 7, 2001.

1.2.1 Regional ITS Architectures

The Florida regional ITS architectures (RITSAs) are derived from the *NITSA*, and define a system of functional requirements and information exchange with planned and existing systems and subsystems in the State of Florida. The RITSAs identify applicable standards, and are tailored to address local situation and ITS investment needs. (Refer to *Section 1.9* for more details on the RITSAs.[4])

1.2.2 Difference between Exempt and Nonexempt ITS Projects

A nonexempt ITS project is defined as one that requires FHWA oversight for projects with funding greater than or equal to $1,000,000 on any interstate or roadway directly serving an interstate using federal aid funding sources such as Surface Transportation Program/Urban (STP/U) funds, National Highway System (NHS) funds, Congestion Mitigation and Air Quality (CMAQ) funds, Interstate Maintenance (IM) program funds, etc. The FHWA is involved in the review and approval of all phases of the project.

An exempt ITS project is one that does not use federal funding, but applies to interstates or roadways serving the interstates. The FHWA's Stewardship Agreement has turn over all oversight activities for the FDOT. The FHWA audits the process or projects that are exempt using process improvement reviews (PIRs) and other mechanisms.

---

[2] PUB.L.NO. 105-178, 112 STAT. 457, *Transportation Equity Act for the 21st Century* (*TEA-21*) (June 1998).

[3] United States Department of Transportation, *National ITS Architecture*, *Version 5.0*. Available online at http://www.iteris.com/itsarch.

[4] More information regarding the Florida District RITSAs is available online at http://www.consystec.com/html/florida/default.htm.

1.3 Program Objectives

The main objective of the SEMP project is to provide guidance for the coordination of ITS communication system deployments throughout the state, advanced traveler information systems (ATIS), roadside sensors, and other ITS devices along major routes, and to provide support for the implementation and maintenance of ITS data warehouses. To standardize the deployment of these services, a user's guide to the application of a comprehensive systems engineering process (SEP) is necessary to ensure that:

- Deployments align with the FDOT's overall mission, goals, and objectives
- Deployments result in effective systems that are fully integrated and coordinated, and that operate seamlessly with other ITS deployments
- Public resources are used with maximum cost efficiency and effectiveness
- Deployments incorporate operation and maintenance plans for the systems that result in reliable, extensible systems
- Guidance and background for Districts to qualify for and continue to receive federal funding for ITS projects is provided
- A simple checklist to follow to qualify for federal funding of ITS projects is provided
- Deployments use a systems engineering management process that is accountable

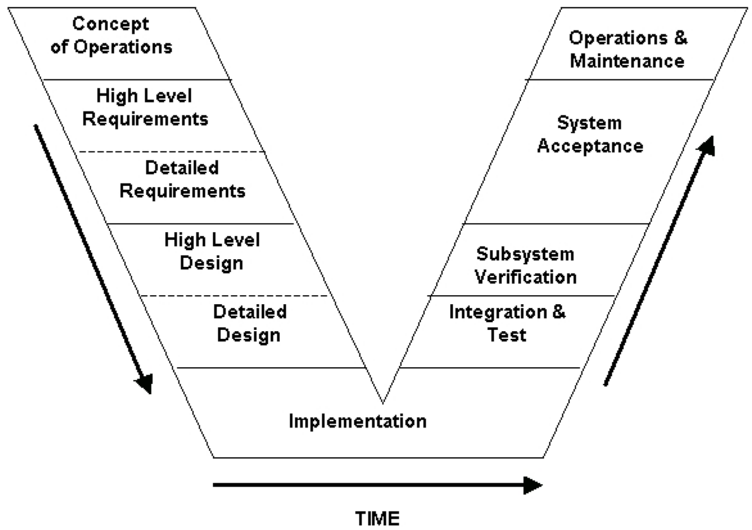
1.4 Program Approach

Florida's Statewide SEMP is a user's guide for the deployment, operation, and maintenance of ITS projects in the State of Florida. The *SEMP* is both informative and instructive, containing process descriptions and checklists to follow; standard document templates for required systems engineering products; and tailored guidelines so a SEP can be cost-effectively adapted based on project size and stakeholder needs.

1.4.1 Program Philosophy

The underlying approach to the application of a SEP is to identify ITS deployment stakeholders, determine their needs, and follow a logical process in defining a system architecture and functional design that can be reviewed and verified to meet stakeholder needs. The key concept in this approach is to identify system requirements, track the requirements to ensure they always link to stakeholder needs, and then verify that the installed system satisfies these requirements. Figure 1.1 shows this in a simplified overview of the approach.

Figure 1.1 – Simplified Systems Engineering Approach – The "V" Diagram[5]



When the SEP is correctly applied, the customer is involved in the entire development process from concept to acceptance and system operation. The systems engineer's primary job is to efficiently allocate the customer's needs to a functional system design whose requirements accurately and adequately define a product that the customer can procure. As the system is being fabricated and integrated, the systems engineer develops the necessary test plans and procedures that are used to verify that all the system requirements are met by the delivered system. Often, the systems engineer will perform independent verification and validation (IV&V) of the delivered system to make sure it meets stakeholder needs. Throughout the development cycle, the systems engineer tracks the requirements back to the user needs to ensure that the system is developed efficiently and cost effectively. Process details needed to provide life-cycle systems engineering are not shown in Figure 1.1. A system's life cycle embraces the system from initial concept to disposal, and requires that the system be designed for ease of maintenance, ease of expansion, easy incorporation of new technology, ease of interfacing with external systems (i.e., interoperability), and ease of operation.

5 United States Department of Transportation, *Freeway Management and Operations Handbook Final Report* (September 2003), FHWA Report No. FHWA-OP-04-003, Electronic Documents Library (EDL) No. 13875. Available online at http://ops.fhwa.dot.gov/travel/traffic/freeway_management.htm.

1.4.2 Project Development Approach

The intent of *Florida's Statewide SEMP* is to define a standard SEP for Florida ITS deployments that is compatible with *Parts 655* and *940* of *Title 23* of the *Code of Federal Regulations* (*CFR*). The focus is on functional requirements, functional design, and functional specifications. The *SEMP* will guide the Florida ITS project engineer through a well-documented process that specifies **what** must be built to satisfy stakeholder needs. It will, in most cases, avoid specifying **how** the system is to be built, so that contractors can propose the most cost-effective solutions possible to satisfy stakeholder needs. It will also define a process for the ITS project team to verify that the delivered system satisfies stakeholder and derived system requirements.

1.5 Program Constraints and Limitations

Florida's Statewide SEMP defines a practical SEP that is tailored to Florida's ITS needs. Many elements of an ideal SEP would not be cost effective or practical for the FDOT to implement. For example, traditional systems engineering considers all aspects of system development, including manufacturing processes and factory testing. The FDOT is not expected to manufacture ITS components, but Districts may become involved in the specification and integration of system components. An immediate constraint to the implementation of a full SEP is the availability of resources. Most FDOT Districts have limited engineering resources trained in the SEP and those that are available are usually fully employed solving information technology (IT) problems. Over time, that may change as the FDOT assumes more responsibility for the development and deployment of Florida's ITS services.

The use of different ITS devices and software throughout the state to perform the same function limits the amount of standardization and configuration control that can be achieved. Over time, this limitation is expected to disappear as a standard equipment list is developed and managed by the FDOT ITS Section.

Nonstandard software and equipment constrain the degree to which different ITS services can communicate and share data throughout the state. Data sharing and the attendant synergy that comes from having access to a common database is limited by compatibility between District communication systems.

Based on surveys and assessments conducted during Phase I of the SEMP project, it was realized that each FDOT District has a slightly different way of operating. Operational differences among the Districts present a challenge in the uniform adoption and employment of *Florida's Statewide SEMP*. One potential solution to this issue is to establish effective training programs early in the *SEMP* deployment phase to standardize the SEP's employment for Florida ITS deployments.

The FDOT is taking steps to remove such constraints and limitations by deploying a standard center-to-center (C2C) communication network, and a standard set of software modules for the operation and maintenance of transportation management centers (TMCs)[6] so that in a few years all Florida Districts will have access to transportation information and the capability to operate other TMCs remotely.

A constraint to the successful statewide deployment of *Florida's Statewide SEMP* is a lack of participation in the *SEMP's* development by key stakeholders. To rectify that, a SEMP Review Committee has been chartered and convened by the ITS Section. The SEMP Review Committee provides across-the-board representation, and coordinates and conducts reviews of *Florida's Statewide SEMP* and its application. The SEMP Review Committee's membership is comprised of the following.

- Principal members include representatives from the following organizations:
  - FDOT Program Management Office (Chair)
  - District Project Engineering
  - District Operations Office
- Advisory members include representatives from the following organizations:
  - ITS contractor(s) selected by the FDOT ITS program manager
  - ITS Florida

6 The set of software modules referred to is the SunGuide[SM] Software System.

1.6 Verifiable Program Goals

It is critical that *Florida's Statewide SEMP* fully satisfy the requirements of the FHWA's *Intelligent Transportation System Architecture and Standards* as defined in *Parts 655 and 940* of *Title 23* of the *CFR* so projects continue to be eligible for federal funding assistance for the procurement and deployment of ITS projects. This *SEMP* prescribes a detailed process for the application of systems engineering practices that are fully compliant with *Parts 655* and *940* of *Title 23* of the *CFR*. Once the process has been standardized and put into practice, the FDOT will need a way to measure its performance so that continuous process improvements can be realized. A baseline that measures how well the FDOT deploys ITS projects was established in the first phase of the SEMP project. Phase I of the SEMP project conducted an appraisal of ITS deployments in Florida using the *Electronic Industries Alliance Interim Standard (EIA/IS) 731.2*.[7] Table 1.1 is a summary of the results of the appraisal conducted in late 2002 and published in *Technical Memorandum No. 1, A Process Review and Appraisal of the Systems Engineering Capability for the FDOT.* [8]

Table 1.1 – Appraisal Results for Florida ITS Deployments[a]

Focus Area	Measured Capability Level
Technical Category	1.07
Management Category	0.47
Environment Category	0.31

[a] The numerical score is derived from a graphic depiction of four subcolumns per capability level. See Tables 3.4 through 3.6 in the referenced document.

[7] Electronic Industries Alliance, *EIA/IS 731.2 – Systems Engineering Capability Model (SECM) Appraisal Method* (December 1998).

[8] Matthews, Traci (PBS&J), *Technical Memorandum No. 1: Phase I – Systems Engineering Management Plan: A Process Review and Appraisal of the Systems Engineering Capability for the Florida Department of Transportation (FDOT)*, *Version 2* (February 2003). FDOT Contract No. C-7772. Available online at http://floridaits.com/system_engineering.htm.

There are six capability levels identified in *EIA/IS 731.1*.[9] Each capability level has practices and attributes associated with process and nonprocess characteristics. These capability level ratings are:

- 0 – Initial (no capability demonstrated)
- 1 – Performed (process steps are recognized and executed)
- 2 – Managed (the process is managed)
- 3 – Dedicated (all participants apply the processes consistently)
- 4 – Measured (results are measured and used for continuous improvement)
- 5 – Optimized (optimal capability demonstrated)

Capability levels are assessed based on performance achievements gained while conducting practices of the focus areas for a given category, thus indicating the FDOT's capability level for that category. As a result of the appraisal, the findings represent an assessment of the implementation level of SEPs within the FDOT as they relate to the *SECM Appraisal Method*. These findings become the benchmark against which process improvements are measured.

Approximately 6 to 12 months after the *SEMP* is accepted and put into practice statewide, a progress assessment survey should be conducted and the results compared to the benchmark. Surveys should be conducted at the District level, and used to direct further training or additional evaluations. This process should be repeated annually until the minimum average score for any focus area is equal to or exceeds 3.0. A goal for each District is to achieve an average *SECM* rating of 4.0 or better within five years.

[9] Electronic Industries Alliance, *EIA/IS 731.1, Systems Engineering Capability Model (SECM)* (December 1998).

1.7 Florida's Statewide SEMP Roles and Responsibilities

The roles and responsibilities for Florida ITS deployment projects are shown in Table 1.2.

Table 1.2 – Roles and Responsibilities for ITS Deployment Projects

Role	Responsibility
FDOT ITS Section	The FDOT ITS Section shall: • Maintain a standard list of approved practices and documentation. • Establish and maintain an approved list of ITS devices using standard interfaces that promote interoperability among District ITS deployments. • Administer a data warehouse. • Provide a C2C communication infrastructure. • Maintain configuration control of statewide TMC software. • Distribute statewide TMC software and support its operation at the District level. • Review District comments. • Host regular meetings of the Change Management Board (CMB). • Maintain a trouble log for statewide ITS deployments.
FDOT District Offices	The FDOT District Offices shall: • Promulgate standard systems engineering practices. • Require the uniform application of systems engineering practices on all District projects. • Periodically review projects for their adherence to *Florida's Statewide SEMP*. • Participate in statewide CMB meetings.
FDOT Project Manager	The FDOT project administrator shall: • Tailor *Florida's Statewide SEMP* to an ITS deployment by creating a project-specific SEMP. • Identify project stakeholders early and determine user needs. • Maintain an active outreach program for the project. • Apply *Florida's Statewide SEMP* guidelines to ITS deployments. • Require contractor adherence to *Florida's Statewide SEMP*. • Require all project participants to be familiar with *Florida's Statewide SEMP*. • Require all purchased ITS devices to conform to the statewide specifications and standards. • Identify and report problems on the statewide trouble report maintained by the ITS Section.
FDOT District Operations Engineer	The FDOT operations engineer shall: • Be familiar with the systems engineering concepts discussed in *Florida's Statewide SEMP*. • Review each District project periodically for compliance with the published *SEMP*. • Ensure that procured items are on Florida's approved list for ITS device compatibility and interoperability.
Contractor	The Contractor shall: • Conform to FDOT documentation requirements. • Conform to FDOT engineering practices. • Actively participate in project reviews.

1.8 Purpose

The purpose of this document is to provide an overview of the standard SEP that is implemented in *Florida's Statewide SEMP*; to provide the context (i.e., the background) for the FDOT SEP; and to cite applicable standards and references to support the standard selected. This document:

- Defines the subprocesses and practices that comprise the systems engineering approach
- Defines criteria for the successful completion of each major activity in the systems engineering approach
- Defines a set of standard practices and technical tools for use in the systems engineering approach
- Establishes a formal process for implementing and improving systems engineering functions

The Institute of Electrical and Electronics Engineers' (IEEE) standard regarding the SEP (i.e., the *IEEE 1220-1998* standard[10]) is the basis for this document, and is tailored by the recommendations and findings from Phase I of the SEMP project.

1.8.1 The IEEE 1220-1998 Standard

The *IEEE 1220-1998* standard specifies the requirements for the SEP and its application throughout a product's life cycle. It does not attempt to define the implementation of each system life-cycle process, but addresses the issues associated with defining and establishing supportive life-cycle processes early and continuously throughout product development. In addition, the standard does not address the many cultural or quality variables that should be considered for successful product development. The standard focuses on the engineering activities necessary to guide product development while ensuring that the product is properly designed.

1.8.2 Definition of Florida's Statewide SEMP

The management plan that provides guidance to Florida ITS engineers is referred to as the *Florida's Statewide SEMP*, and it provides the tools necessary to develop, deploy, and maintain an ITS project. The *SEMP* is broad in nature and each project is expected to generate its own project SEMP based on the guidelines contained in *Florida's Statewide SEMP*.

[10] Institute of Electrical and Electronics Engineers, *IEEE Std. 1220 – Standard for Application and Management of the Systems Engineering Process* (December 1998).

1.9 The NITSA and Florida's SITSA – What Do They Mean and What Are They Used For?

In the context of ITS, the "architecture" graphically describes what a system does and generally how it does it. It identifies the processes to be performed by subsystems and components, and defines the flows of information and the interfaces between them. The following paragraphs come from the *FDOT Draft Rule 940 Procedures*,[11] published separately from this *SEMP*.

The *NITSA* provides a common framework for planning, defining, and integrating ITS services. It reflects the contributions of a broad cross-section of the ITS community, such as transportation practitioners, systems engineers, system developers, technology specialists, and consultants, over a nine-year period. The architecture defines:

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

The *NITSA* by its nature is generic and needs to be adapted to a particular need. The *Florida Statewide ITS Architecture* (*SITSA*) *and Standards*[12] is a tailoring of the *NITSA* for the State of Florida, and some market packages, data flows, and other elements of the *NITSA* are augmented with additional data unique to Florida's ITS needs. Some ITS Districts in Florida have developed RITSAs that further tailor the *Florida SITSA* to meet more regional needs. The *NITSA*, *SITSA*, and RITSAs are used in the SEP as a reference to make sure all the key elements of the project architecture are addressed and are compatible with the state and regional architectures. An ITS architecture derived from a regional or local plan is not sufficient by itself to implement an ITS deployment. *Florida's Statewide SEMP* is needed for that.

---

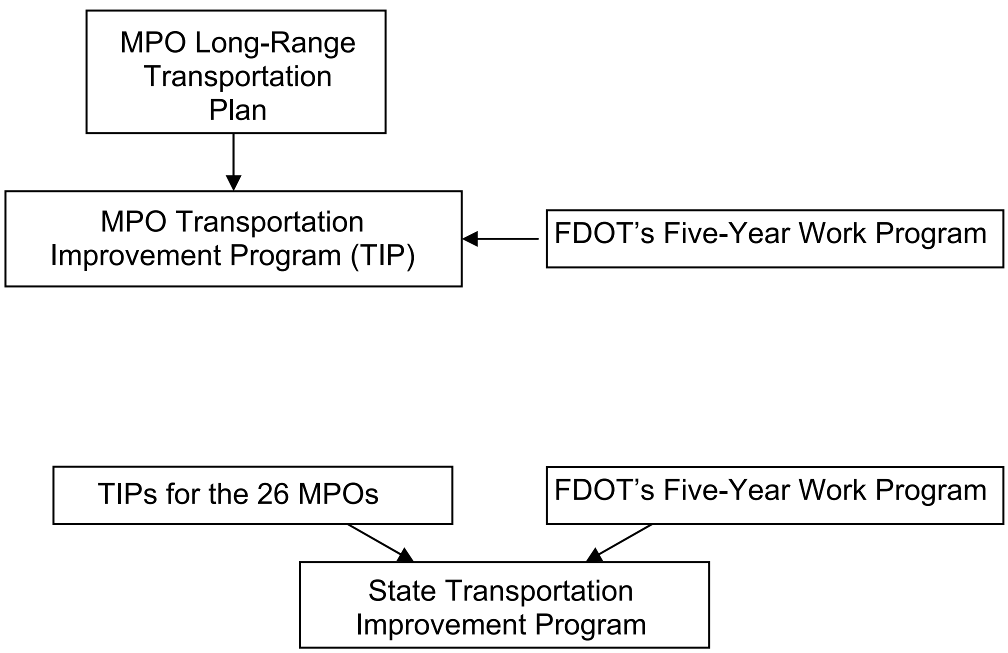
[11] Quigley, Diane E. (PBS&J), *Florida Department of Transportation Draft Rule 940 Procedures in Florida* (December 2003). FDOT Contract No. C-7772. Available online at http://floridaits.com/rule_940_implementation.htm.

[12] Ice, Ron (Jaffe Engineering), *Florida Statewide Intelligent Transportation System Architecture and Standards* (June 2000). FDOT Contract No. C-7354. Available online at www.consystec.com/html/florida/.

1.9.1 How the Planning and Funding Process Takes Place

There are 26 metropolitan planning organizations (MPOs) in Florida and each one develops a long-range plan for improving transportation in their metropolitan area. Each plan identifies potential projects, estimates a budget for each project, and considers what resources the FDOT has allocated for the next five years. The 26 MPO plans are reviewed and coalesced into the five-year state transportation improvement plan (STIP). This process is depicted in Figure 1.2.

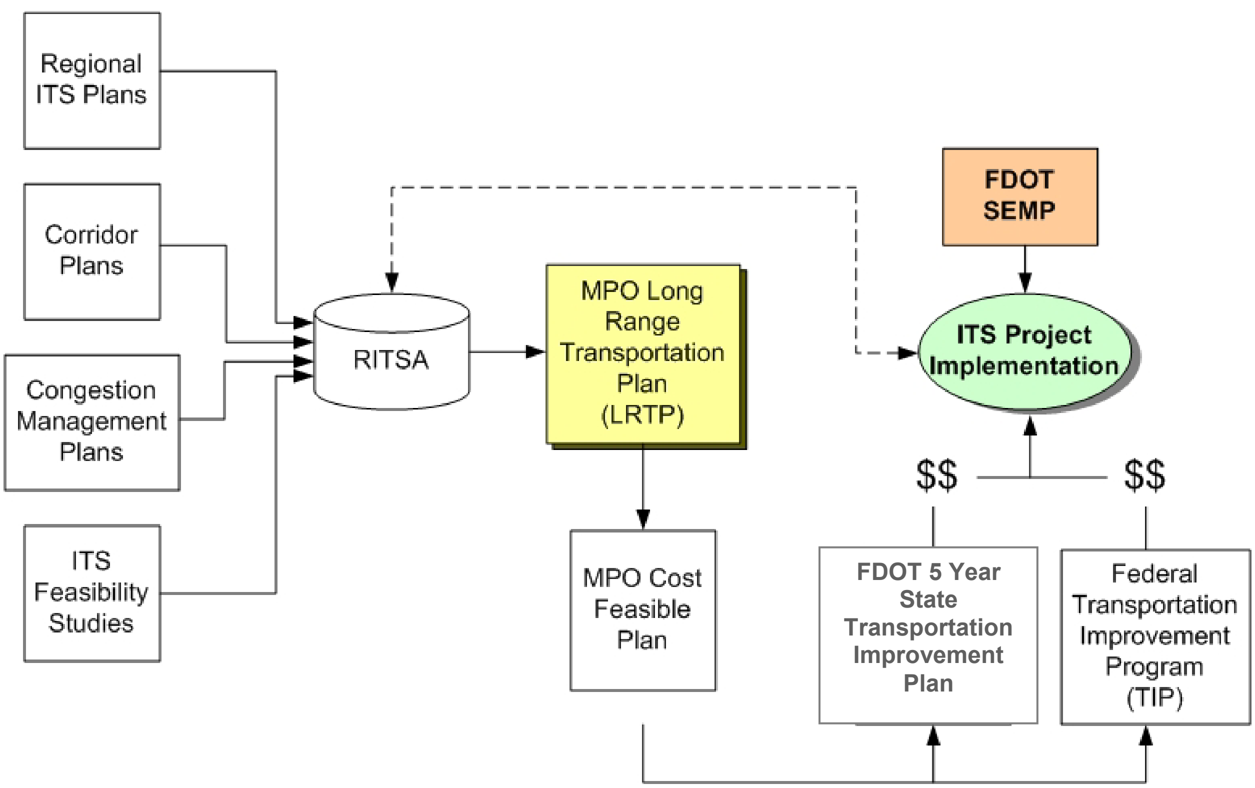
Figure 1.2 – Transportation Planning and Funding Process



1.9.2 How a Project is Selected for Implementation

The recommended projects the District project engineer will implement can come from various sources. The FDOT's *Ten-Year ITS Cost Feasible Plan*, referred to as the *Ten-Year ITS CFP*,[13] only addressed the Florida Intrastate Highway System (FIHS) limited-access corridor ITS projects. Additionally, the District may have identified other projects or needs from regional ITS implementation plans, corridor studies, or ITS feasibility studies; or a MPO may have identified the need as part of a their congestion management system (CMS). Ultimately, the ITS projects recommended in these studies need to be in the RITSA if they are to be developed further. A high-level screening of each project solution is the part of the SEP used to determine if the project, or what portions of the project, are in the RITSA. The "should be" process being implemented by the FDOT, shown in Figure 1.3, depicts the projects evolving from the RITSA sequence of projects being integrated into the MPO long-range planning processes.

Figure 1.3 – Project Identification and Funding Process



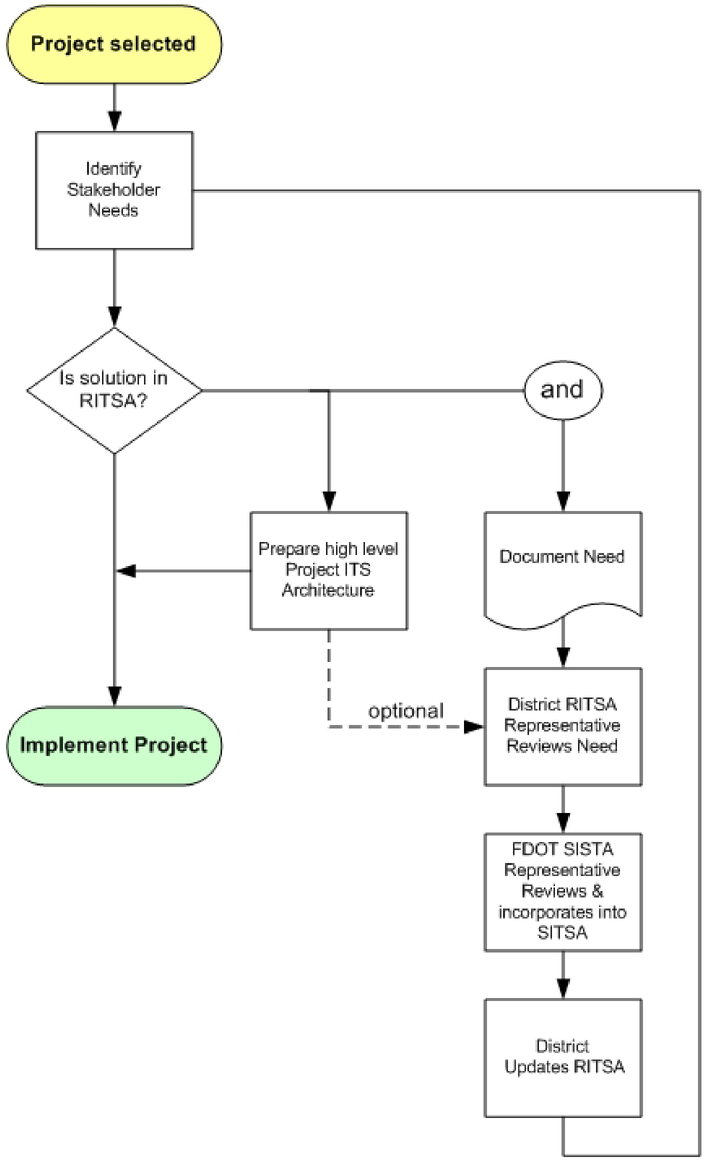
[13] Florida Department of Transportation, *Ten-Year ITS Cost Feasible Plan* (May 2004). Available online at http://www.dot.state.fl.us/trafficoperations/its/its_default.htm under the Online Documents link.

Each MPO is expected to implement an ITS chapter in their long-range transportation plan (LRTP) that identifies the recommended solutions from the various studies referenced earlier (i.e., the *Ten-Year ITS CFP*, ITS feasibility studies, CMS needs, etc.). The LRTP evaluates the solutions with respect to RITSA high-level projects or project portions, and funds those high-priority ITS projects in the MPO cost feasible plans (CFPs). Eventually, these projects are entered in the Transportation Improvement Program (TIP), which is required for federal funding, and into the FDOT's *Five-Year Transportation Improvement Plan* for each metropolitan area, as required for state funding.

When a project reaches the TIP and the FDOT Five-Year Work Program,[14] the District ITS project engineer steps in and begins to refine the project/problem, its requirements, and the solutions. The District ITS project engineer prepares a detailed analysis of the alternatives to determine what systems consistent with requirements of the overall ITS program best meet the requirements, have the best benefit-to-cost ratio, or are most feasible. At this point, the ITS project engineer should ask, "Is the solution(s) to the problem that we have identified in the regional architecture? If so, what portions are we implementing? What market packages, subsystems, data flows etc.?" Ideally, the RITSA design is based on the same stakeholder needs and requirements, therefore the problems and solutions the ITS project engineer is defining have likely been conceived of and addressed in the RITSA. If the solutions are not in the RITSA, then the ITS project engineer should follow the process depicted in Figure 1.4.

[14] More information regarding the FDOT's Five-Year Work Program is available online at http://www2.dot.state.fl.us/programdevelopmentoffice/wp/.

Figure 1.4 – The RITSA Update Process Flow



If not all the solutions to the identified stakeholder needs are found in the current RITSA, the District ITS project engineer must:

- Document the need to modify the RITSA and pass that on to the appropriate District RITSA representative, who will then pass it on to the ITS Section's *SITSA* administrator.
- Prepare a high-level project architecture to be incorporated in the RITSA later. The high-level project architecture is required by the SEP if it is not already a part of the RITSA.

If a high-level project architecture is developed locally, the District ITS project engineer is requested to forward a copy to the ITS Section's *SITSA* administrator for incorporation in the *SITSA* and the appropriate RITSA.

1.9.3 Application of Florida's Statewide SEMP

Florida's Statewide SEMP enables the ITS project engineer to manage a project using systems engineering principles and methods to maximize the quality of the system being implemented while minimizing the budget and schedule required for its completion. At the start of a particular ITS project, the project engineer will use *Florida's Statewide SEMP* to create a project SEMP tailored for that particular project.

It is assumed that an ITS deployment project will be managed by an FDOT engineer who may be assisted by an FDOT project team and general engineering consultant (GEC) staff. In *Florida's Statewide SEMP*, the FDOT person responsible for managing the technical aspects of an ITS project deployment is the ITS project engineer. A test director, who would be responsible for planning and executing the system's IV&V program, should assist the ITS project engineer. The ITS project engineer may be assisted by engineers expert in the various specialty engineering disciplines that include software, reliability, quality management (QM), and configuration management (CM).

1.10 How to Use This Document

Florida's Statewide SEMP is both a reference manual and a guide that the ITS project engineer will use to structure a project that meets the *Rule 940* requirements. The FHWA's *Rule 940* is discussed in *Appendix D* to provide a background for the SEP that is described in this document. The classic SEP is presented step-by-step in *Section 3* and provides a good overview on all the elements of systems engineering throughout a system's life. *Section 3* is not oriented towards ITS specifically, as the principles discussed can be applied to any problem needing a systems solution. Read this section to understand the systems engineering principles behind the FHWA's *Rule 940*. Templates of typical plans are provided in the appendices.

Section 4 provides a description of the SEP as it is applied to ITS and should be followed by the ITS project engineer to implement an ITS project in Florida. If the ITS project engineer is familiar and conversant with the *NITSA* and *SITSA*, the engineer should start with *Section 4* and use it as an ITS project implementation guide.

The *NITSA* and *SITSA* are discussed in *Appendix C* along with automated tools that help the ITS project engineer define a regional or local ITS architecture when needed. Other appendices provide information on using the ITS tools, creating the standard documentation specified by the SEP, using process checklists, and other useful information.

To help the ITS engineer develop standard SEP documentation, several templates are provided in the appendices. *Appendix B* is a definition of common SEP terminology; *Appendix C* provides a description of the SEP when using the *NITSA* Turbo Architecture software tool.[15] A full description of the remaining appendices are described in *Section 3.14*.

It is recommended that *Florida's Statewide SEMP* be used to create a project-specific SEMP to be maintained in a three ring, loose-leaf notebook and used as the standard guide for the ITS project engineer and staff when deploying ITS projects. A Web-based version will also be provided through the *i*Florida project Web site.[16]

1.11 Systems Engineering Review Checklist Form

The SunGuide[SM] ITS Checklist (SIC) Form[17] has been developed to help District engineers ensure that planning for Florida ITS projects that are expected to use federal funds meets the requirements of *Rule 940*. This checklist is contained in *Appendix A* as a form to be filled out and submitted with the request for federal funds. Not all items on the SIC need to be answered in the positive as long as the reason for a negative or null response is documented in the space provided. Additional pages of comments can be attached to the form as needed.

[15] More information regarding the United States Department of Transportation's (USDOT) Turbo Architecture software application is available online at http://mctrans.ce.ufl.edu/featured/ from the Center for Microcomputers in Transportation (Mc*Trans*).

[16] More information regarding the FDOT's *i*Florida Surface Transportation Security and Reliability Information System Model Deployment project is available online at http://www.iflorida.net/.

[17] SunGuide is a service mark of the Florida Department of Transportation.

This page is intentionally left blank.

CHAPTER 2 – REFERENCED DOCUMENTS

2. Document Applicability

The following documents, of the exact issue shown, form a part of this document to the extent specified herein. In the event of a conflict between the documents referenced herein and the contents of this document, this document shall be considered the superseding requirement.

Referenced Documents	
IEEE Standard 1220-1998 *Standard for Application and Management of the Systems Engineering Process* Approved December 8, 1998	The Institute of Electrical and Electronics Engineers, Inc. 345 East 47th Street New York, NY 10017-2394
Issue Paper *The Approved Systems Engineering Approach for ITS Deployments Along Florida's Limited-Access Corridors* *Version 13.0*[19] May 6, 2003	Florida Department of Transportation Traffic Engineering and Operations Office Intelligent Transportation Systems (ITS) Section 605 Suwannee Street, M.S. 90 Tallahassee, FL 32399-0450
Technical Memorandum No. 1 *Phase 1 – Systems Engineering Management Plan: A Process Review and Appraisal of the Systems Engineering Capability for the Florida Department of Transportation (FDOT)* *Version 2* February 20, 2003	Florida Department of Transportation Traffic Engineering and Operations Office Intelligent Transportation Systems (ITS) Section 605 Suwannee Street, M.S. 90 Tallahassee, FL 32399-0450
ISO/IEC 15288 *Systems Engineering – System Life Cycle Processes*[20] October 2002	ISO Central Secretariat International Organization for Standardization (ISO) 1, Rue de Varembe, Case Postale 56 CH-1211 Geneva 20, Switzerland

[19] PBS&J, *Issue Paper – The Approved Systems Engineering Approach for ITS Deployments Along Florida's Limited-Access Corridors*, *Version 13* (May 2003). FDOT Contract No. C-7772. Available online at http://floridaits.com/system_engineering.htm.

[20] International Organization for Standardization/International Electrotechnical Commission, *ISO/IEC 15288 – Systems Engineering – System Life Cycle Processes* (October 2002).

This page is intentionally left blank.

CHAPTER 3 –
THE SYSTEMS ENGINEERING PROCESS DEFINED

3. The Systems Engineering Process

3.1 Overview of the Systems Engineering Process

The purpose of this chapter is to provide an overview of the standard systems engineering process (SEP) that is implemented in *Florida's Statewide SEMP*; to provide the background for the FDOT's SEP; and to cite applicable standards and references to support the standard selected. This chapter:

- Defines the subprocesses and practices that comprise the systems engineering approach
- Defines criteria for the successful completion of each major activity in the systems engineering approach
- Defines a set of standard practices and technical tools for use in the systems engineering approach
- Establishes a formal process for implementing and improving systems engineering functions

The *IEEE 1220-1998* standard is the basis for this chapter*,* and is tailored by the recommendations and findings from Phase I of the SEMP project.

3.1.1 The IEEE 1220-1998 Standard

This standard specifies the requirements for the SEP and its application throughout a product's life cycle. It does not attempt to define the implementation of each system life-cycle process, but addresses the issues associated with defining and establishing supportive life-cycle processes early and continuously throughout product development. In addition, the standard does not address the many cultural or quality variables that should be considered for successful product development. The standard focuses on the engineering activities needed to guide product development while ensuring that the product is properly designed.

3.1.2 Feedback

Emphasis in the FDOT SEP is placed on stakeholder involvement in the entire system's life cycle and depends on feedback from the stakeholders at key points to improve the system. Much emphasis is placed on gathering user needs at the beginning of the project and using the concept of operations (ConOps) process to refine those needs into a consensus agreement among the stakeholders for the system's top-level functional requirements. As those requirements are translated into system functional and performance requirements, frequent reviews with the stakeholders will help refine the interpretation and understanding of the requirements in the context of the system as a whole. During the technical development phase, early demonstrations of system functionality are used to confirm that the requirements were clearly understood and properly interpreted. A characteristic of the SEP used by *Florida's Statewide SEMP* is that there will be no surprises when the system is deployed. It may not be perfect and it may not meet all of the requirements, but the *SEMP* provides a process to identify and correct deficiencies, and incorporate product improvements while supporting the system's operation throughout its life cycle.

3.1.3 Integration Planning During the Design Phase

The role of test director is identified and the position staffed at the beginning of a project based on *Florida's Statewide SEMP*. The test director's job is to plan for the integration and testing (I&T) of the system as it is being designed but not to actually do the integration – that job is usually performed by the contractor. Emphasis is placed on requirements testability, meaning that when a design requirement is stated, the method of proving that the system meets the requirement is also defined. Another characteristic of *Florida's Statewide SEMP* is that it recommends a separate group be established to independently verify and validate that system requirements are met. The job of this IV&V group is to find all the system faults and get them fixed before the system is deployed. It is often said that the main job of the IV&V engineers is to break the system before the stakeholders do.

3.1.4 Requirements Verification During Integration

Another important characteristic of the SEP's modified Vee model used in *Florida's Statewide SEMP* is the phased integration of the system with demonstrations of system functions during integration. This phased integration provides opportunities for early verification and validation that system requirements are met, as well as opportunities for the stakeholders to showcase and potentially deploy early functionality. This is sometimes referred to as "showcasing early-winners."

3.2 The FDOT's Systems Engineering Process Objective

The FDOT's statewide SEP ensures that the mission, goals, and objectives of the FDOT and other stakeholders are met by ensuring that ITS deployments result in a fully integrated, seamless, and coordinated multimodal system that uses public resources in the most cost-effective and efficient manner possible. In addition, employing a standard SEP provides agencies with an invaluable tool for maximizing the likelihood of a project's successful deployment.

3.2.1 Application of the Systems Engineering Process

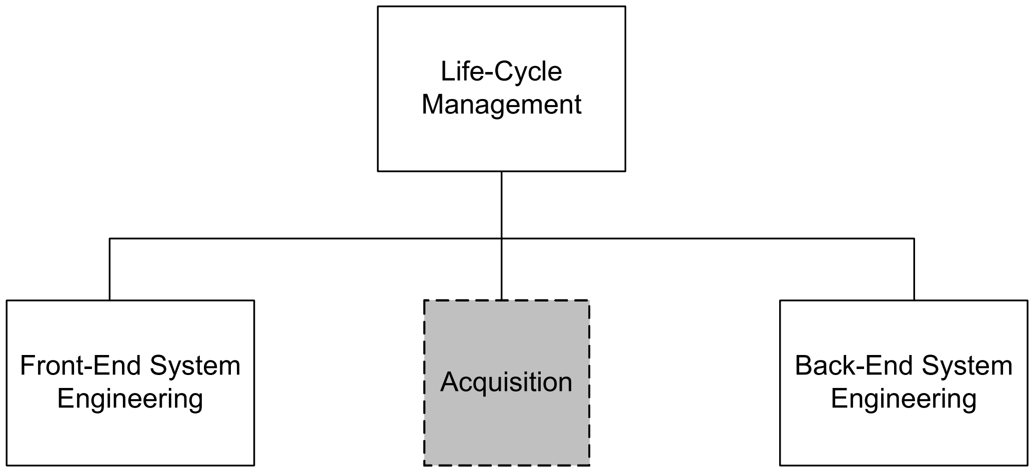
The FDOT's SEP enables an ITS project engineer to manage a project using systems engineering principles and methods to maximize the quality of the system being implemented while minimizing the budget and schedule required for its completion. At the start of a particular ITS project, the project engineer should use *Florida's Statewide SEMP* as a guide to detail what aspects of the statewide SEP will be applied to a particular project. This is specified by a SEMP tailored for that particular project. *Florida's Statewide SEMP* contains guidance, templates, and helpful hints to tailor the statewide SEP.

It is assumed that an ITS deployment project will be managed by an FDOT engineer who may be assisted by an FDOT project team and GEC staff. In the FDOT SEP, the FDOT person responsible for managing the technical aspects of an ITS project deployment is the ITS project engineer. A test director, who will be responsible for planning and executing the system's IV&V program, should assist the ITS project engineer. The ITS project engineer may also be assisted by engineers expert in the various specialty engineering disciplines that include software, reliability, and QM and CM processes.

3.3 Standard Systems Engineering Description

The basic SEP structure can be thought of as three parts with an over-arching fourth part. One part is the front-end work that will define the problem, propose solutions, select an optimal solution, define requirements based on user needs, and produce the necessary documentation that will govern the implementation of an ITS project. The acquisition part is supported by the front-end documentation augmented by a budget analysis that estimates what the system should cost to acquire. It supports the development of the necessary documentation needed to acquire the system or contract with someone to build it. The back-end part is the work the systems engineering staff accomplishes to verify and validate that the system solution meets all the requirements defined in the front-end. The back-end continues to support the deployment and operation of the system throughout its life cycle. The over-arching part is the management processes and controls that ensure the system meets stakeholder expectations throughout its life cycle. This concept is shown in Figure 3.1.

Figure 3.1 – Systems Engineering Parts



- Requirements Identification and Analysis
- Architecture definition
- Design Synthesis
- Requirements Management
- System Specifications

- Scope of Services
- System Specification
- Contract Terms and Conditions
- Budget Analysis

- Integration and Test Concept
- Integration Testing
- Acceptance Testing
- Configuration Management
- Quality Control
- Operations and Maintenance Planning
- Training
- Operations Support
- Continuous Product Improvement

The acquisition process is shown in gray in Figure 3.1 because the process is supported by the SEP, but is not the SEP's prime responsibility.

3.3.1 The Front-End Systems Engineering Process

The most important rule in this SEP is to fully and completely understand the ITS project stakeholders' needs and their expectations. The SEP devotes a great deal of effort to this end and has many checkpoints in the process to verify that system development is on track meeting stakeholder expectations. The very first step for the ITS project engineer is to identify the participating agencies, and their roles and responsibilities (i.e., the stakeholders), then to understand the problem and define its boundaries. This SEP advocates that once the stakeholder agencies are identified and the problem is bounded, the applicable portions of the RITSA can be identified and requirements can be specified.

Typically, the general problem to be solved by the ITS deployment is already identified in a CFP and it is up to the ITS project engineer to solve it. However, classical systems engineering dictates that a clear understanding of the problem be achieved by working with the stakeholders.

3.3.2 Problem Definition – The Five Whys?

The *NITSA* and *SITSA* offer many ingredients that can be blended together to create an ITS architecture that might represent a possible solution to a perceived ITS need. Before solutions can be selected and applied, the ITS project engineer must first understand the problem to be solved, and then define the problem based on stakeholder needs. During the process of defining the problem, the ITS project engineer will be able to determine who the real stakeholders are in the project. The best way to understand the problem is to hold interviews with the people identified as key stakeholders. Obviously, the person managing the project budget is a key stakeholder and has that budget because certain other people in the organization asked for money to meet a perceived need. This is the person to start with. The "five whys" technique has proven effective in the past.[21] This is simply a process of asking "why?" at least five times in a row to detect the root cause or meaning of a particular problem or situation. The following is an example of this technique when the ITS project engineer asks a key stakeholder to state their need.

Stakeholder: I need a towel.

ITS Project Engineer: **Why** do you need a towel?

Stakeholder: To wipe up the water on the floor.

ITS Project Engineer: **Why** is there water on the floor?

Stakeholder: It is dripping from the ceiling.

ITS Project Engineer: **Why** is the water dripping from the ceiling?

Stakeholder: A pipe has broken in the ceiling.

ITS Project Engineer: **Why** did the pipe break?

Stakeholder: Because the heat is turned off at night and the water in the pipe froze.

ITS Project Engineer: **Why** is the heat turned off at night?

Stakeholder: It's a company policy that has never been changed since we moved from Florida to Alaska.

[21] The "five whys" is a guideline suggested by Masaaki Imai for troubleshooting problems. It was made popular as part of the Toyota Production System in the 1970s. More information is available in *Kaizen: The Key to Japan's Competitive Success*, by Masaaki Imai, published by McGraw-Hill (1989).

The root cause of the problem is that a policy change is needed. The other answers were symptoms of the problem.

This process can be followed either through one-on-one interviews with key people in the stakeholder organizations or through working group meetings with the stakeholders. The primary product is a set of stakeholder needs defined according to stakeholder expectations of what the system must be able to do.

3.3.3 Requirements Analysis – Defining Solutions

Once the problem has been clearly stated as a set of stakeholder needs, the next step is to define a solution or set of solutions, and pick the optimal one. This subprocess in the SEP is called the **requirements analysis**, or the analyzing of alternative design solutions. The ITS project engineering team shall perform requirements analysis for the purpose of establishing what the system shall be capable of accomplishing; how well system products shall perform in quantitative, measurable terms; the environments in which system products will operate; the human/system interface requirements; the physical/aesthetic characteristics; and constraints that affect design solutions. The market needs, requirements, and constraints are derived from stakeholder expectations; project and enterprise constraints; external constraints; and high-level system requirements.

The RITSAs can assist in identifying **functional requirements**. However, the ITS architecture does not deal with other types of requirements, such as the performance requirements that are required as part of the SEP presented in the *IEEE 1220-1998* standard.

The first task is to define customer expectations in terms of functional requirements, performance requirements, the operational environment, and constraints. This includes:

- What the customer wants the system[22] to accomplish (i.e., the functional requirements)
- How well each function should be accomplished (i.e., the performance requirements)
- The natural and induced environments in which the system product operates or may be used
- Constraints, such as funding; cost or price objectives; schedule; technology; nondevelopmental and reusable items; design characteristics; hours of operation per day; on-off sequences; external interfaces; and specified existing equipment, procedures, or facilities related to life-cycle processes

[22] The term "system" refers to system products, life-cycle processes, and desired quality factors.

A wealth of analysis has already been accomplished in developing the *SITSA* and the RITSAs based on the *SITSA*. The *SITSA* and RITSA databases contain a great deal of data that can be used to develop a project ITS architecture that helps meet the FHWA's *Rule 940* requirements for regional and statewide ITS compatibility.

The *SITSA* and, thus, any derived project architecture based on the *SITSA*, does not explicitly list the regional stakeholder requirements and constraints. However, these requirements and constraints are implicitly considered in the selection and customization of the market packages called for in the RITSAs. In addition, the RITSA database includes the identification of legacy and planned systems; organizational responsibilities; and existing and planned external interfaces that can be considered as constraints in project development and deployment. The *SITSA* development team identified this information based on existing regional and corridor deployments; existing ITS architectural documentation; and articulation of stakeholder needs in the regional workshops.

This information needs to be verified as part of a project's development, and can be used in the identification of the requirements and constraints of a project. Completing this task will also satisfy *Rule 940* requirements for the identification of participating agencies' roles and responsibilities.

The way to find out what the functional requirements are is to use a structured process called "hierarchal input-process-output," or HIPO, analysis.

3.3.3.1 Identify Components of the Regional Architecture Being Implemented

The HIPO analysis has been used since the 1970s to design computer software systems and is an easy way to specify a functional architecture to satisfy user needs. The HIPO analysis has also been called the "black box" design technique because if you can define what the inputs are to a process and what outputs are needed, then you don't really care what goes on inside the box. When this is done in a hierarchal top-down fashion, the resulting functional design states the problem solution clearly and generates the system functional requirements in a linked, modular manner. The HIPO analysis described is essentially the process of defining ITS market packages. Since RITSAs have been developed for all the ITS Districts in Florida, the ITS project engineer needs to select from the already-defined market packages that appear to address the problem to be solved.

3.3.3.2 ITS Market Packages

The ITS market packages represent collections of subsystems and terminators that exchange information, which is illustrated with architecture flows in the market package diagrams, to do a specific service. The market packages are customized to represent the operational concept for service delivery specific to a region. Each subsystem or terminator in a market package diagram is labeled with both its generic *NITSA* name and the name of the local stakeholder that participates in the specification of the customized market package. In this way, the market package identifies the information exchange (using architecture data flows) between specific stakeholder elements in the region to affect a particular service or set of services. This is also a good source to generally identify who the stakeholder agencies are. The ITS project engineer will still need to identify individuals within each stakeholder agency to involve in the project.

The *SITSA* development team decided on the use of the market packages from the *NITSA* as a starting point for the architecture analysis, rather than starting from the user service requirements, which are actual statements of user needs that do not specify a particular architectural implementation. This is because the team concluded that the abstract concepts of user services, while invaluable for the systems engineers that developed the *NITSA*, are generally too abstract for the majority of stakeholders in a limited time setting.

The *FDOT SITSA* final report[23] mentioned that the above approach should not be taken to understate the importance of the logical architecture. The logical architecture is crucial to understanding the physical architecture in sufficient detail to develop interface standards, and to understand the underlying processes that explain what a physical subsystem does. The report indicated that these details are important to standards developers and project designers at the plans, specifications, and estimates (PS&E) stage, but are less important to regional stakeholders and large investment decision makers who are responsible for RITSA requirements and decisions.

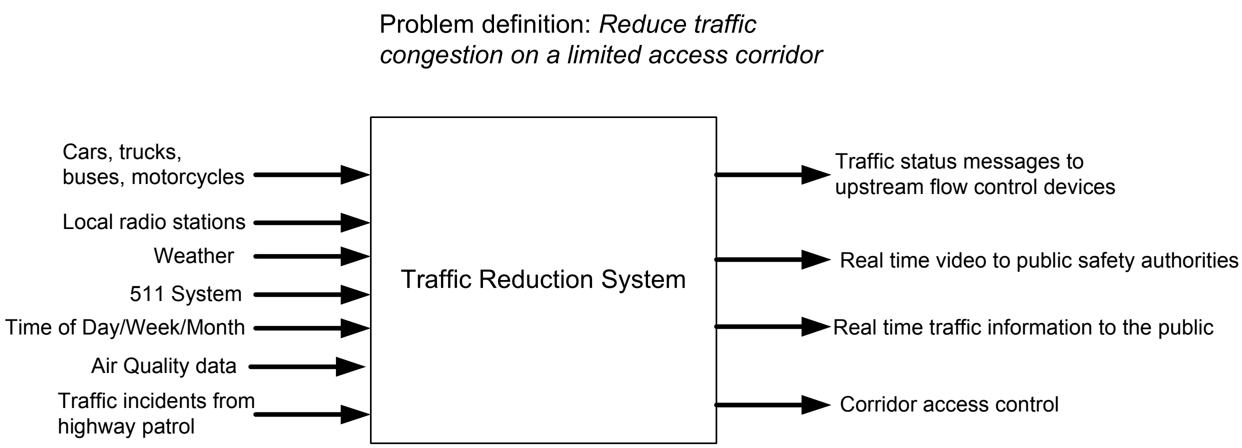
3.3.3.2.1 Example of Selecting Market Packages from the RITSAs

The first step for the ITS project team is to take a sheet of paper or use a computer-based drawing package and draw a large box. Label the box with the name of the ITS project being designed or the project name. Next, draw arrows going into the box on the left side, and label the arrows by the data or information you feel is necessary for the black box to process into outputs. Next, draw arrows leaving the box on the right side, and label each arrow with what data or information it represents. The first box drawn is at the systems level and must be at a very high level. This view of the system is sometimes called the 10,000-foot view.

[23] Jaffe Engineering and Development Industries and Kimley-Horn and Associates, Inc., *FDOT Statewide ITS Architecture and Standards* (September 2002). Available online at http://www.dot.state.fl.us/trafficoperations/its/its_default.htm under the ITS Architectures, Standards and Modeling link and the ITS Architectures sublink.

Figure 3.2 is a simple example of a top-level HIPO analysis diagram for a system to solve a traffic congestion problem on a limited-access corridor. At the top level, the process can be called the system name, but in all lower level boxes, the process should be stated as an action to be taken (i.e., an action verb). In the example shown, the inputs to the process are identified. An input to the process is defined as anything the process cannot control but needs to process to generate a solution to the problem. For example, local radio stations provide input to the process because they influence driver behavior, as does a traveler information system like 511. Weather also effects traffic congestion and air quality may be an indicator of traffic congestion depending on the weather.

Figure 3.2 – Example of a Top-Level HIPO Analysis Diagram for a Traffic Reduction System



Other inputs do not cause the problem, but are a by-product of it and can be used to make decisions in the processing of the data to solve the problem. Hence, air quality sensor data is an input in conjunction with weather to determine when specific traffic congestion mitigation strategies should be employed. Historical traffic congestion data based on the time of day, day of the week, and month also can serve as valuable external input in the process of predicting when traffic congestion will occur and the development of proactive congestion mitigation strategies.

The system outputs are designed to control access to the limited-access corridor; to influence driver behavior; to alert authorities of incidents so they can respond quickly and clear any obstacles in the highway; and to provide safety to the public. This process at the highest level is defining the system boundaries. According to the *IEEE 1220-1998* standard, the ITS project engineering team needs to define their system boundaries, including:

- Which system pieces are under the design control of the enterprise and which fall outside its control
- The expected interactions among system pieces that are under design control, and external and/or high-level and interacting systems outside the system boundary

The ITS architecture will help in this effort by identifying the system pieces provided as part of the project, and by identifying other existing or future pieces that will interact with the pieces provided.

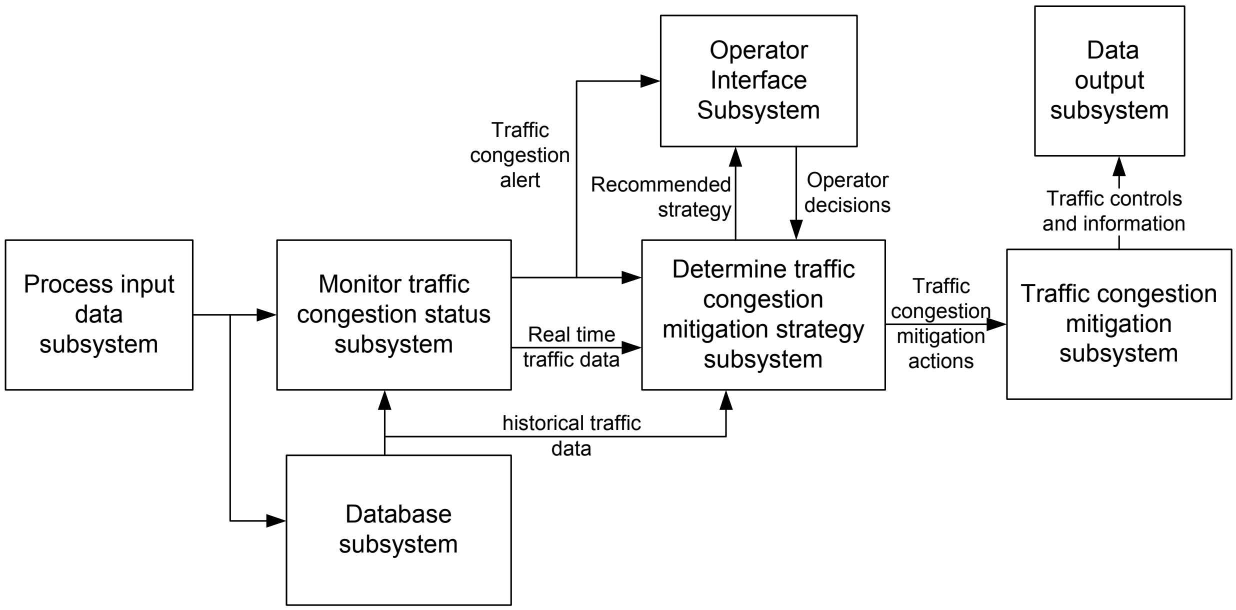
To illustrate an example using the *SITSA*, assume that FDOT District 6 wanted to solve a problem for a particular stretch of Interstate 95 (I-95) near Miami. From Figure 3.2, the ITS project engineer can see that portions of the highway must be watched, and traffic flow detected and measured. In reviewing the market packages for District 6 found at the *Florida SITSA* Web site, the ATMS04 (Freeway Control) market package appears applicable, as do the ATMS05 (High-Occupancy Vehicle [HOV] Lane Management), ATMS06 (Traffic Information Dissemination), and ATMS08 (Traffic Incident Management System) market packages.[24] Figure 3.2 indicates a need for a traveler information dissemination subsystem, but there is no market package defined for District 6. However, the ATIS1 (Broadcast Traveler Information) market package is specified for District 4 and this could serve as a model to create one for District 6.

For each of these market packages, a set of architecture data flows are already defined, albeit somewhat generically. These data will help the ITS project engineer to specify a more detailed ITS project architecture as the example continues.

The next step is to arrange the subsystem processes (i.e., market packages) that must take place within the system process box to take the input data and generate the required output data. This hypothetical system black box that solves a traffic congestion problem on a limited-access corridor is expanded in Figure 3.3 to subsystem processes. There are many ways to define and arrange the subsystem processes, and it is recommended that the functional boxes defined minimize the data flow between them. By doing this, the ITS project engineer can define a modular system architecture that is relatively simple to implement and maintain. The subsystems shown in Figure 3.3 represent a system that gathers real-time data; relies on historical data to predict when congestion might occur or what will probably happen as traffic density increases along the corridor or weather deteriorates; and generates mitigation strategies for the operator to select from and implement. Even at this second level of decomposition, the ITS project engineer can start to infer system characteristics that may influence the stakeholders' perceived needs. This decomposition process of the functional processes continues until the system can be clearly stated functionally, and the inputs and outputs (e.g., interfaces) are well defined.

[24] It is recommended that the project architecture use the term "subsystem" whenever a market package uses the term "system" to avoid confusion with the ITS project being implemented.

Figure 3.3 – Subsystem Level of the HIPO Analysis Process



Keep in mind that the applicable RITSA may not have all the market packages needed to specify a project architecture and the HIPO process will help fill in the missing pieces. Once the ITS project architecture has been defined, the ITS project engineer will need to work with District stakeholders to update the RITSA database to incorporate the additional market packages created for the project.

3.3.3.3 Interface Definition

The ITS project engineering team needs to define the functional and design interfaces to external and/or high-level and interacting systems, platforms, humans, and/or products in quantitative terms. Mechanical, electrical, thermal, data, communication-procedural, human-machine, and other interfaces are included. The project ITS architecture includes information that allows the engineering team to identify information flows between the project functions and subsystems.

There are two types of interfaces in an ITS architecture.

- **Architecture Flows** – These represent information exchanged between architecture pieces in the functional view of the system. Architecture flows are the primary tool that is used to define the regional and project ITS architecture interfaces in Turbo Architecture. (Refer to *Appendix C*.)

- **Data Flows –** Data flows represent data flowing between functions within a system, or between a system piece and an external interface point.[25] A data flow is shown as an arrow on a data flow diagram and is defined in a data dictionary entry in the logical architecture. Data flows are modeled in the logical architecture view of the *NITSA* and are aggregated to form high-level architecture flows in the physical architecture view of the *NITSA*. Data flows can be the source of requirements because a subsystem requirement would be to provide general kinds of data (i.e., functional requirements) or a specific type of data within a certain period of time (i.e., performance requirements).

In the HIPO analysis process, data flows are aggregated depending on the level of system decomposition. For example, at the subsystem functional block diagram level, the data flows are identified generally but at a lower level view, such as the element functional level, where the individual data may be identified. For detailed project development tasks, these flows need to be traced to their component data flows that provide a higher level of detail of the flows between the subsystems. As data is identified, a separate document should be maintained that maps all the data names to their aggregate data flows. This document is called the data dictionary.

3.3.3.3.1 ITS Interface Standards

The United States Department of Transportation (USDOT) leads an effort to develop standards for the interfaces between ITS subsystems and the *NITSA* database that can be used to identify applicable ITS standards for a project. The *NITSA* database contains a mapping of each architecture flow to ITS standards that support the flow. With the ITS architecture created as described above, an output that is available from the *NITSA* is a set of applicable standards for each data flow. This set serves as a starting point for the specification of ITS standards as part of the project specifications. Documenting the set of applicable standards and testing procedures, along with the rationale for the standards selected for the project, satisfies the *Rule 940* requirement for "identification of applicable ITS standards and testing procedures."

It should be noted that the Turbo Architecture software produces a list of ITS interface standards that are being developed by the ITS community and includes a data dictionary, message sets, and communication protocol standards. Additional enabling standards are used with ITS that include commercial off-the-shelf (COTS) communication media and applicable general-purpose data communication standards, such as standards for fiber optic technology, the Transmission Control Protocol/Internet Protocol (TCP/IP), and cellular radio. These standards are also applicable to the corridor ITS deployments, although not included in the list.

[25] The *NITSA* refers to this as a terminator.

At this point, if the ITS project engineer reviewed the system functional architecture shown in Figure 3.3 with the stakeholders, a logical question that might be asked by the stakeholders is why there appears to be no provision for a map-based display of the traffic along the corridor. This may not have been stated previously as a user need or it may be a component of one of the subsystems, yet the ability for the ITS project engineer to diagram the functions and get stakeholder feedback can further define user needs and expectations, and ultimately the system requirements. An excellent tool to use in conjunction with the HIPO analysis is the ConOps document.

3.3.3.4 The Concept of Operations

The role of the ConOps in the SEP is to describe how the delivered system will work in lay terms from the stakeholders' viewpoint. Typically, system operation is described from four stakeholder viewpoints: the system user, the system operator, the system maintainer, and the system manager. The purpose of the ConOps is to convey to the stakeholders a vision of how the system will be implemented and what impact it will have on them. When done properly, the ConOps is instrumental in achieving consensus among the stakeholders on the vision for the system and a common set of user needs. This needs to be done very early in the SEP and is usually accomplished within two months of the project start. The process of developing the ConOps is described in detail in *Section 5* and a template is provided in *Appendix R* of this document for the ITS project engineer's use.

3.3.3.4.1 Regional Concept of Operations

A ConOps can be developed on any level, from project specific to statewide and regional, and its focus will shift depending on what level it is developed for. The white paper titled *Regional Concepts of Operations for Transportation System Management and Operations*[26] discusses the ConOps from the regional level and focuses more on operating relationships among the regional ITS stakeholders. The type of ConOps described *in Florida's Statewide SEMP* is project level, and should be used to refine stakeholder expectations and obtain consensus on a project vision.

3.3.3.4.2 Project Concept of Operations

A project ConOps states the problem to be solved and creates a hypothetical system composed of functions that are used to describe how the system will address stakeholder needs. Writing the ConOps is an iterative process that generates functional requirements from the system level down to the component level and below, and verifies them with the stakeholders. Often, portions of the ConOps become outreach materials to educate the public and others about what is being designed and built.

---

[26] Federal Highway Administration, *White Paper: Regional Concepts of Operations for Transportation System Management and Operations, Discussion Draft 2.1* (February 2003).

The recommended way to write a project ConOps is to do it while the HIPO analysis process is being used to decompose the system functions into lower level components. Create the first two levels of the system HIPO analysis using the RITSA market packages as much as possible, and then start writing a description of how it works, what data flows into the boxes, and what happens to the data to become outputs. As the writing progresses, the engineer will think of additional functional details that needs to be addressed. Often, the engineer will discover that certain inputs are needed external to the system to support a process within the system, or the engineer will reach an impasse in describing system operation based on the functions as they are diagrammed. This could be an indication that the system boundaries and constraints have not yet been fully identified, and a further examination of the RITSA or even the *SITSA* is needed, or the conceptual architecture needs to be discussed with the stakeholders.

The ConOps document states the problem to be solved, assumes a system has been built that meets all the stakeholders' needs, and describes its operation based on the four stakeholder viewpoints. It is important that this section be read and understood by all the stakeholders, so it should be easy to read and understand, and should not be too technical. What has been effective in previous ITS ConOps documents is to write the description of the system operation like a short story that takes the time line of the story and shows it from the viewpoint of each of the four stakeholder groups. Consideration should be paid to the system operation description in all conditions of the environment in which it will operate.

3.3.3.4.3 Utilization Environment

The SEP requires that the ITS project engineering team define the utilization environments for each of the operational scenarios. All environmental factors, natural or induced, that may affect system performance should be identified and defined. Factors are identified that ensure that the system minimizes the potential for human or machine errors, or failures that cause injurious accidents or death, and that pose minimal risk of death, injury, or acute chronic illness, disability, or reduced job performance of the humans who support the system life cycle.

As applied to ITS, defining utilization environments could include, for example, operating a traffic management system under normal, incident, and emergency conditions; or operating a transit system during weekdays, weekends, or special events.

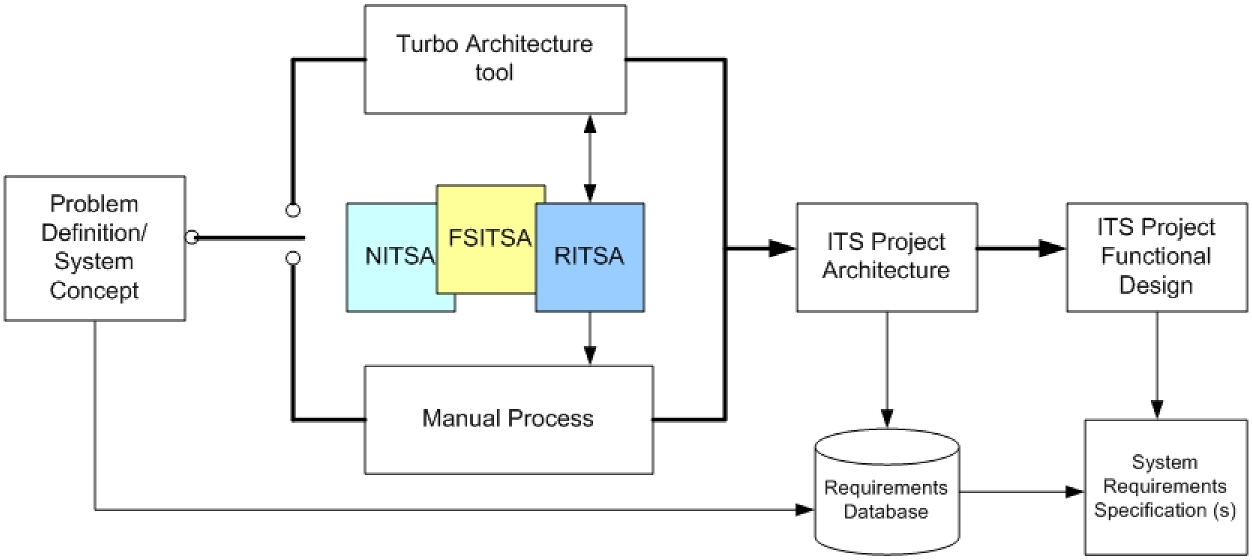
After the hypothetical system's operation is described, the HIPO analysis diagrams and the appropriate market packages that were integrated in the diagrams are presented to show how the system can do everything that was described in the story line. In the SEP, these diagrams are called system functional block diagrams. This will tie system capabilities to system functional components in the mind of the stakeholder. Stakeholder expectations may also be tempered by cost considerations, so the ConOps is an excellent way to convey the expected cost of the system to the stakeholder and offer alternative approaches. The cost data in a ConOps is called a rough order of magnitude (ROM) price and is intended to give a ballpark estimate of what the system could cost. Typically, a ROM reflects a higher price than what is really expected so as not to surprise the stakeholders as the system is procured. Finally, the ConOps should restate the user needs and the top-level system requirements that satisfy those needs. Typically, the draft ConOps is reviewed at the SRR. (Refer to *Section 4.6.1.1.3* of this document.)

At this point in the SEP, the ITS project team should have a solid understanding of the stakeholder needs and a functional system architecture that satisfies that need along with a set of top-level system functional requirements linked to the stakeholder needs. The next step in the SEP is to use the *SITSA* to structure the system architecture and select the applicable standards.

There are two ways to proceed in defining the ITS project architecture. The FHWA's *Rule 940* requires that it use the *NITSA* and other derivations of it to develop the ITS architecture for a particular project. (Refer to *Appendix D*.) If the ITS project is based on a RITSA, the architecture may already be defined and easily used for the particular project. If the ITS project engineer has a good understanding of the SEP and is comfortable using the Turbo Architecture tool, then the specific ITS project architecture can be refined using the *SITSA* with the Turbo Architecture software along with data flows, requirements, and other elements of ITS design. However, if the fundamental structure of the *NITSA* and RITSA, and the use of the Turbo Architecture software tool are not well understood, it is better for the ITS project engineer and the team to continue with the SEP manually to arrive at a definition of the project's ITS architecture, and subsequent functional design and attendant requirements. Both ways reference the *NITSA*, the *Florida SITSA*, and the appropriate RITSAs to satisfy *Rule 940*. (Refer to *Appendix D* for a discussion of *Rule 940*.) This is graphically depicted in Figure 3.4. (Also, refer to *Section 1.4.1*.)

Refer to *Appendix C* for a guide to using the Turbo Architecture software tool to define a local ITS project architecture. The following sections discuss the manual process of developing the ITS project architecture.

Figure 3.4 – Defining the ITS Project Architecture



3.3.3.5 Constraints

Constraints can be categorized as project constraints, enterprise constraints, and external constraints.

3.3.3.5.1 Project Constraints

The SEP requires that the ITS project engineering team identify and define project and enterprise constraints that impact design solutions. Project constraints may include:

- Approved specifications and baselines developed from prior applications of the SEP
- Updated engineering and technical plans
- Team assignments and structure
- Automated tools availability or approval for use
- Control mechanisms
- Required metrics for measuring technical progress

3.3.3.5.2 Enterprise Constraints

Enterprise constraints may include:

- Management decisions from a preceding technical review
- Enterprise general specifications, standards, or guidelines
- Policies and procedures
- Domain technologies
- Established life-cycle process capabilities
- Physical, financial, and human resource allocations to the technical effort

3.3.3.5.3 External Constraints

The ITS project engineering team should also identify and define external constraints that impact design solutions or the implementation of SEP activities. These constraints include:

- Public and international laws and regulations
- The technology base
- Industry, international, and other general specifications, standards, and guidelines
- Human-related specifications, standards, and guidelines
- Human availability, recruitment, and selection
- Competitor product capabilities

Technical Memorandum No. 3.2: Technology Review[27] prepared as part of the ITS Corridor Master Plans for the five FIHS limited-access corridors addresses various implementation, and operations and maintenance issues associated with market packages that could assist in this task.[28] The *NITSA* documentation also addresses some of the constraints, issues, and risks associated with ITS deployments. (Refer to *Appendix C*.)

[27] Hadi, Mohammed (PBS&J), *Technical Memorandum No. 3.2: Technology Review – ITS Corridor Master Plans for Florida's Principal FIHS Limited-Access Corridors*, FDOT Contract No. C-7772 (July 2002).

[28] More information regarding the FDOT's ITS Corridor Master Plans is available online at http://www.dot.state.fl.us/trafficoperations//its/its_default.htm under the ITS Deployment link.

3.3.3.6 Design Synthesis – Further Refining the Requirements

The purpose of the SEP's synthesis subprocess is to define design solutions and subsystems that satisfy the verified functional requirements. The synthesis subprocess as described in the *IEEE 1220-1998* standard translates the functional architecture into a design architecture that provides an arrangement of system elements; their decomposition; internal and external interfaces; and design constraints. The design architecture also includes the requirements traceability and allocation matrices, which capture the allocation of functional and performance requirements among the system elements. Design architecture definitions should be stored in the integrated database, along with trade-off analysis results, design rationale, and key decisions to provide traceability of requirements up and down the architecture.

In the ITS architecture language, synthesis is translating the ITS logical architecture into a physical architecture. As part of the *NITSA's* physical architecture development, the ITS logical architecture process specifications (PSpecs)[29] were assigned to equipment packages, and these equipment packages were assigned to subsystems and market packages, satisfying the synthesis subprocess requirements. In addition, the *NITSA* groups logical architecture data flows into physical architecture flows. Traceability between logical and physical elements; trade-off analyses; rational; and data dictionaries are also provided as part of the *NITSA*. The *SITSA* customized the *NITSA* physical architecture elements and interfaces to Florida conditions. As part of the synthesis subprocess, there is a need to examine the project architecture developed based on the *SITSA* to determine if there is a need to modify this architecture based on the identified requirements and constraints.

There is also a need as a part of the synthesis to analyze alternative configurations and technology options to meet the requirements. This analysis may result in recommending modifications to the project architecture developed based on the *SITSA*. The alternative architecture configurations need to be analyzed, identifying their strengths and weaknesses, and the best alternative selected based upon selection criteria created by or discussed with the project management team.

Technology choices can play a key role in identifying an ITS project's physical architecture. For example, the ATMS02 (Probe Surveillance) market package will look different, depending on the technology selected (e.g., transponders, license plates, cell probes, transit vehicle automatic vehicle location [AVL] systems, or telematics global positioning system [GPS] devices.

It is recommended that the technology options for key elements in the project be considered, and that the selection of those technologies be based upon selection criteria created by or discussed with the project management team. This selection is done within the context of the project's requirements (i.e., the chosen technologies meet the requirements based on the selection criteria). Another set of alternatives that could be examined at this step is procurement options.

---

[29] The PSpecs may be thought of as system functions.

Completion of the analysis of alternate architectures, technologies, and procurement options satisfies the *Rule 940* requirements for:

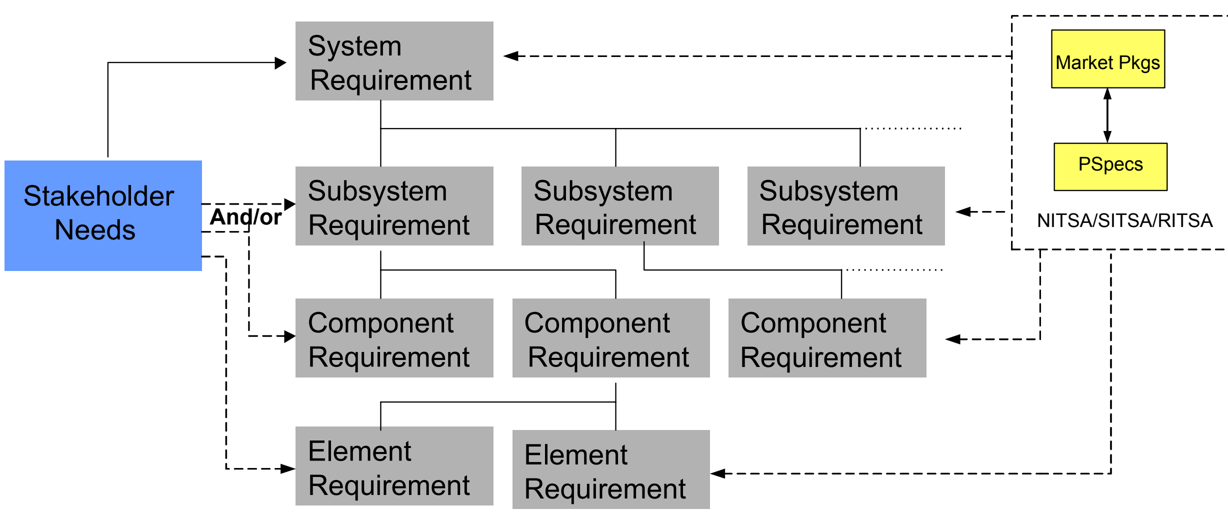
- Analysis of alternative system configurations and technology options to meet requirements; and
- Procurement options.

The design process will include creating specifications for the project design. The specifications can draw heavily from the requirements definition, and the contribution to these requirements made by the statewide or regional ITS architectures as described above. Specifications of individual systems, or the overall project specifications, may provide more detail than the requirements definition.

3.3.4 Tracking Project Requirements and the Use of a Database

A key characteristic of the SEP is the traceability of all requirements back to stakeholder needs. Further, requirements must be defined in a hierarchal manner. The hierarchal term used in requirements traceability is the "parent-child" relationships of the requirements. A design rule followed to achieve a workable design specification is for there to be only one parent requirement for any child requirement; however, there may be many children of a single parent requirement. This hierarchy of requirements is illustrated in Figure 3.5.

Figure 3.5 – Requirements Hierarchy and Parent-Child Relationships



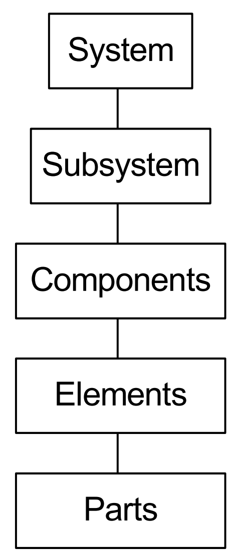
In Figure 3.5, a single system requirement is shown to have three children – the subsystem requirements. The first subsystem requirement has two children that are called component requirements; the two element requirements are children of the second component requirement. Figure 3.5 also shows the relationship that the user needs and the *NITSA*, *SITSA*, or RITSA requirements have with the ITS project requirements. User needs may be expressed at any level of the design hierarchy and it is often a challenge to the systems engineer to correctly link a user need to a system requirement without violating the one parent-many children rule. The *NITSA*, *SITSA*, or RITSA-driven requirements are already hierarchal, and it is easier to link those to the ITS project requirements. The importance of a structured design and a hierarchy of functions must be understood in order to manage the requirements and track them properly.

3.3.4.1 Functional Design Hierarchy[30]

There are many sizes of systems in the world from biological systems, ecological systems, weather systems, and solar systems to such smaller systems, such as computer processor units. The *IEEE 1220-1998* standard views a system as an element of a larger system, and the challenge for the ITS project engineer is to understand the boundary of the system, which is the focus of the development effort, and the relationships and interfaces between this system and other systems.

The SEP recognizes only one "system" in the development of a project's ITS architecture and design, so a ITS project's system is composed of related pieces (i.e., subsystems and components) and their interfaces. Figure 3.6 lists a hierarchy of names for the pieces making up a system. This generic system hierarchy is a key concept within the *Florida Statewide SEMP* because it ties the system architecture; specifications and drawing trees; system breakdown structure (SBS); technical reviews; and configuration baselines together. Many pieces within the system hierarchy can be considered a "system" by the classical definition, but actually represent subsystems within the system hierarchy. Most projects should not require any further decomposition of the system than the five levels shown in the figure. Note that there is only one "system" in the ITS project, but that system may be composed of many subsystems. Each subsystem may have many components, as the components may have many elements, and the elements may have many parts.

Figure 3.6 – System Name Hierarchy



[30] This information has been adapted from *Section 1.4* of the *IEEE 1220-1998* standard.

Since the focus of the SEP is to specify the system by its functional requirements and to let the contractor provide it, there is little reason to define the system in any more detail than the part level. The functional requirements are developed in a hierarchal manner by defining the system level requirements, and then taking each requirement and allocating it to the next lower level of functional decomposition. Requirements must be maintained and tracked throughout the project to ultimately prove that the system that is delivered and installed meets all the requirements. Requirements must be maintained in a database and identified in a logical way to identify each requirement uniquely. The SEP recommends that requirements be related to the system functional hierarchy in a way that provides an easy means to relate a requirement to its place in the system hierarchy. This is done in the requirements database by using a unique letter-number combination.

3.3.4.2 Design Verification

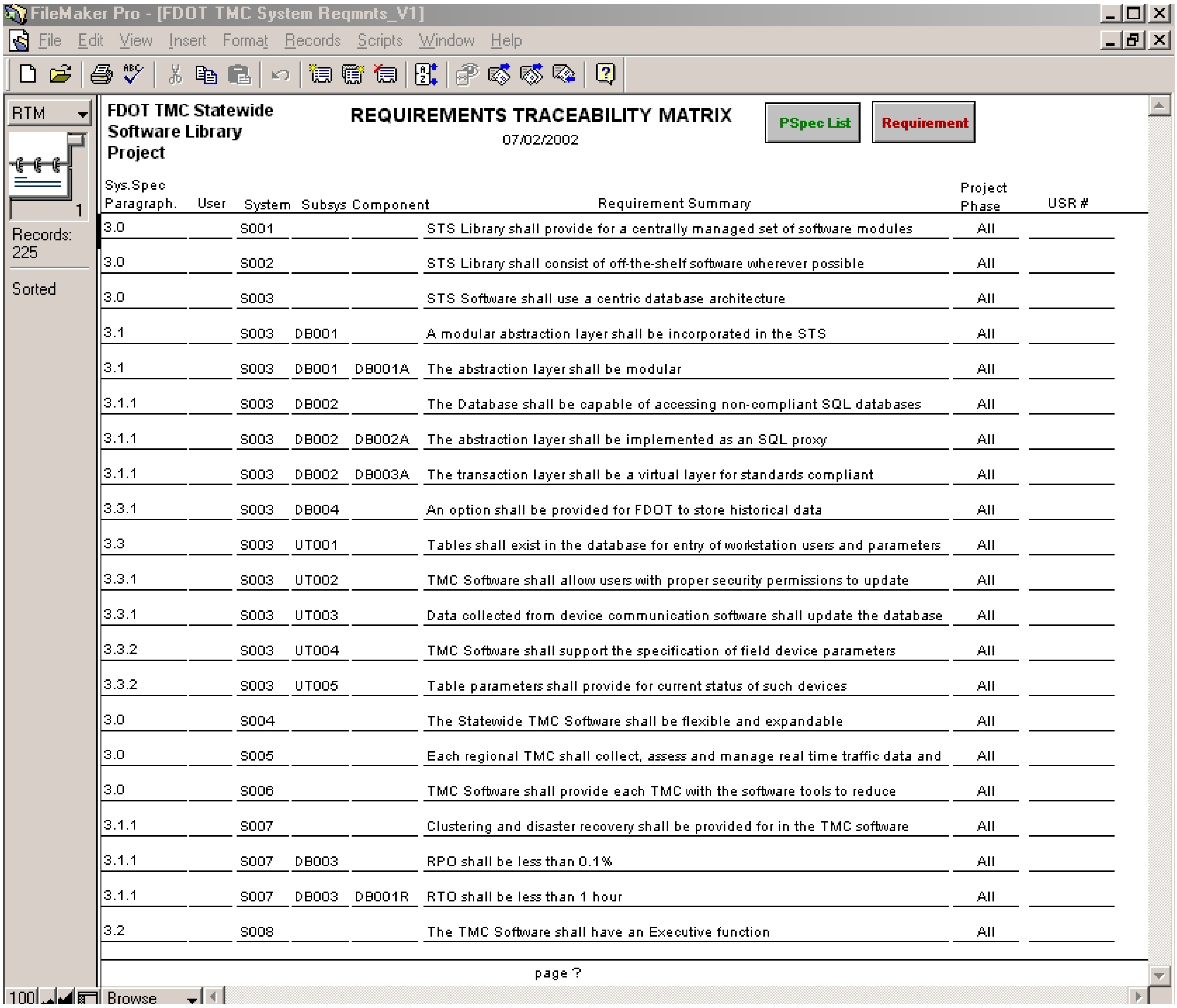
As part of the SEP, the ITS project engineer will perform design verification to ensure that the design architecture's lowest level requirements, including derived requirements, are traceable to the verified functional architecture, and that the design architecture satisfies the validated requirements baseline.

At the design verification stage, the physical architecture is examined to determine if it satisfies all the requirements and constraints of the project. Automated tools that map the physical architecture elements to the requirements and constraints are not commercially available, but a requirements tracking tool is available that supports the manual mapping of the requirements hierarchy to the stakeholder needs.

3.3.4.3 Requirements Database and Tracking Tool

A SEP convention is that all stakeholder requirements be designated by the letter A and a unique three-digit number (e.g., A001). The letter A is used because this is the very top of the requirements hierarchy. All system-level requirements are identified by the letter S and a unique three-digit number (e.g., S001). As mentioned previously, there may not be a one-for-one correspondence between A-level requirements and S-level requirements, but from the S level down to the lowest piece of the system, the parent-child relationship is followed rigorously. The use of a hierarchal requirements identification scheme makes it easy to manage the requirements using a database such as the requirements traceability matrix (RTM) example shown in Figure 3.7.

Figure 3.7 – Sample Requirements Traceability Matrix



FDOT TMC Statewide Software Library Project

REQUIREMENTS TRACEABILITY MATRIX

07/02/2002

Sys.Spec Paragraph.	User	System	Subsys	Component	Requirement Summary	Project Phase	USR #
3.0		S001			STS Library shall provide for a centrally managed set of software modules	All	
3.0		S002			STS Library shall consist of off-the-shelf software wherever possible	All	
3.0		S003			STS Software shall use a centric database architecture	All	
3.1		S003	DB001		A modular abstraction layer shall be incorporated in the STS	All	
3.1		S003	DB001	DB001A	The abstraction layer shall be modular	All	
3.1.1		S003	DB002		The Database shall be capable of accessing non-compliant SQL databases	All	
3.1.1		S003	DB002	DB002A	The abstraction layer shall be implemented as an SQL proxy	All	
3.1.1		S003	DB002	DB003A	The transaction layer shall be a virtual layer for standards compliant	All	
3.3.1		S003	DB004		An option shall be provided for FDOT to store historical data	All	
3.3		S003	UT001		Tables shall exist in the database for entry of workstation users and parameters	All	
3.3.1		S003	UT002		TMC Software shall allow users with proper security permissions to update	All	
3.3.1		S003	UT003		Data collected from device communication software shall update the database	All	
3.3.2		S003	UT004		TMC Software shall support the specification of field device parameters	All	
3.3.2		S003	UT005		Table parameters shall provide for current status of such devices	All	
3.0		S004			The Statewide TMC Software shall be flexible and expandable	All	
3.0		S005			Each regional TMC shall collect, assess and manage real time traffic data and	All	
3.0		S006			TMC Software shall provide each TMC with the software tools to reduce	All	
3.1.1		S007			Clustering and disaster recovery shall be provided for in the TMC software	All	
3.1.1		S007	DB003		RPO shall be less than 0.1%	All	
3.1.1		S007	DB003	DB001R	RTO shall be less than 1 hour	All	
3.2		S008			The TMC Software shall have an Executive function	All	

The identification of requirements below the system level is tied to the names of the system pieces as much as possible. The subsystem requirements are identified by a unique two-letter and three-digit number, for example, a closed-circuit television (CCTV) subsystem requirement might be labeled TV001.

Component level requirements retain their subsystem identification letter code but add a suffix letter after the unique three-digit number (e.g., the video wall component of the CCTV subsystem requirement might be TV001W).

The element level requirement mimics the component level identification scheme, but adds a unique single-digit number to the requirement identification after the suffix (e.g., a switching element of the CCTV subsystem's video wall component might be identified as TV001W1. At this level, it is not practical to devise a logical naming convention and keep the identification code relatively simple.

Finally, a part level requirement can be identified by adding a lower case letter to the end of the element level requirement identification number (e.g., a control protocol requirement for the switch element of the video wall component of the CCTV subsystem might be TV001W1a).

By using a requirements identification scheme as suggested above, it will be easy to sort and group requirements to support analysis and trace requirements back to stakeholder needs.

3.3.4.4 Types of Requirements

The SEP identifies only three types of requirements that should be considered, including:

- **Functional Requirements –** These requirements describe *what* is to be done, not *how*. An example of a functional requirement is: "The CCTV subsystem shall provide real-time video to the regional transportation management center (RTMC)." Most of the requirements that the ITS project engineering team will specify will be functional.

- **Performance Requirements –** These requirements describe how well a part of the system is to perform in quantifiable terms. An example of a performance requirement might be that the geographic information system (GIS) map shall be updated every five seconds. Avoid specifying performance requirements unless it is necessary to meet a stakeholder need or support another system part because it is more time consuming and thus more costly to verify a performance requirement than a functional requirement. Further, it constrains the contractor's options in providing a cost-effective system solution to meet the requirements.

- **Physical Requirements –** These are *how to* requirements that are very specific and are dictated by the environment that the system will operate in. An example is a legacy communication infrastructure system that will connect to the system being designed. The requirement might be that the CCTV pan-tilt-zoom (PTZ) control commands conform to the TCP/IP because the legacy communication backbone is Gigabit Ethernet®.[31] Other examples of physical requirements are height and weight requirements, or environmental requirements that specify the range of temperatures and humidity the system must operate in.

[31] Ethernet is a registered trademark of Xerox Corporation.

3.3.5 Performance Measures

The SEP requires that the ITS project engineering team define system effectiveness measures that reflect overall stakeholder expectations and satisfactions. The measures shall be related to project stakeholder goals and objectives.

The best way to determine performance measures for the ITS project being specified is to ask each stakeholder how they will know the system is effectively addressing their stated needs. Try to quantify the performance measures so they can be measured. For example, if a stakeholder says the system will be considered effective if traffic incidents are decreased, the ITS project engineer should probe further to determine if a 5 percent reduction the first year and a 10 percent reduction the second year would be a good indicator of performance. Further, a "traffic incident" probably needs to be defined in a little more detail to avoid ambiguity later when traffic incidents are being counted.

Besides asking the stakeholders, there are a number of other sources that can assist in this regard. The *NITSA's ITS Performance and Benefit Study*[32] presents qualitative and quantitative performance measures that are mapped to national goals for ITS development and deployment. These goals include the improvement of operational efficiency, mobility, safety, economic productivity, and the environment.

Florida's ITS Strategic Plan[33] outlines the recommended development of an ITS plan; the deployment priorities for ITS goals and objectives; and the performance measures to be reported. These performance measures can be categorized as:

- Safety measures
- Protection of public investment measures
- Interconnected transportation system measures
- Travel choice measures

Performance measures were also recommended in *Technical Memorandum No. 3.3*,[34] which was developed as a part of the ITS Corridor Master Plan project referenced previously. These performance measures include mobility- and safety-related performance measures, as well as agency performance measures. Each of these measures was derived from the goals and objectives statements used to summarize the needs, issues, problems, and objectives for ITS deployments on the FIHS corridors, or to support a hierarchy of national performance measures.

---

[32] Lockheed Martin Federal Systems, Odetics Intelligent Transportation Systems Division, *ITS Performance and Benefit Study* (June 1996).

[33] Florida Department of Transportation, *Florida's Intelligent Transportation System Strategic Plan – Final Report* (August 1999).

[34] Shaw, Terrel (PBS&J), *Technical Memorandum No. 3.3: ITS Program Performance Measures – ITS Corridor Master Plans for Florida's Principal FIHS Limited-Access Corridors*, FDOT Contract No. C-7772 (June 2002).

3.3.5.1 Measures of Effectiveness and Measures of Performance

Two ways to evaluate how well a system design meets its requirements is to define measures of performance (MOPs) and measures of effectiveness (MOEs). Definitions of the two measures are provided below, based on the *IEEE 1220-1998* standard:

- **Measures of effectiveness** are the metrics that a customer will use to measure satisfaction with products produced by the technical effort.

- **Measures of performance** are the engineering performance measures that provide the design requirements needed to satisfy MOEs. There are generally several MOPs for each MOE.

When functional system requirements are defined and low-level requirements are allocated to subsystems, components, and elements of the system, the ITS project engineer should select or specify requirements that are testable. Testable requirements are MOPs that can be traced to stakeholder requirements and their MOEs.

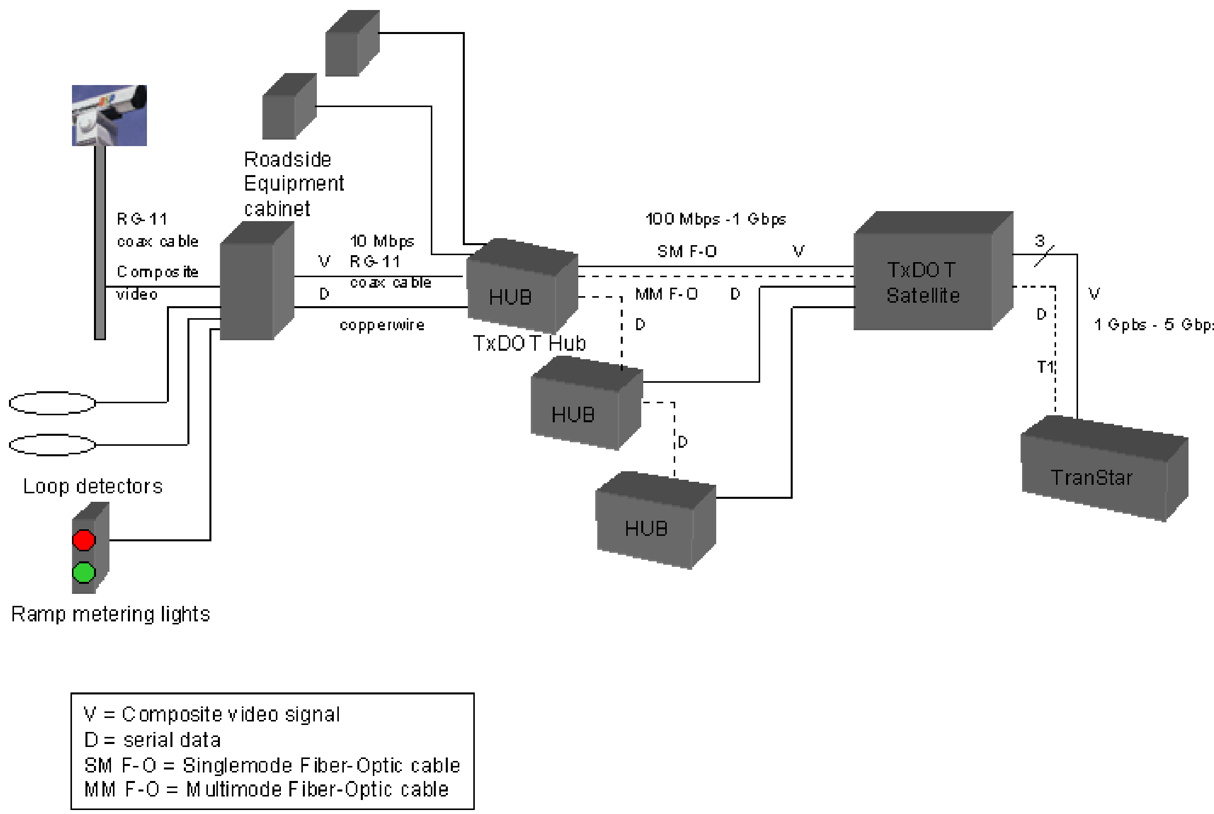
3.3.6 Analysis of Alternate Designs and Technologies – Cost-to-Benefit Analysis

It has been often stated that there are many different ways to accomplish a goal, with some ways taking longer or costing more than others. This is very true of ITS projects. There are different ways to combine the market packages, different ways to group functional parts of the system, and different technologies that deliver the same data. The FHWA's *Rule 940* requires that alternate approaches be considered before selecting the best one to proceed with. The SEP calls this a cost-to-benefit analysis of alternate designs and technologies. What may appear to be the cheapest solution to implement may cost more in the long run and would not be a good candidate for implementation.

3.3.6.1 Alternate Conceptual Architectures and Technologies Example

The *SITSA* and the derived RITSAs have a basic architecture defined by the market packages. The HIPO analysis processed described earlier creates a project architecture based on those market packages and data flows, but there is some latitude left to the ITS project engineer to arrange them to optimize data flows or provide system redundancy, for example. Often, how the functional subsystems are arranged will affect the system's cost and may depend on technology to make the configuration work. A simple example would be a traffic surveillance system that delivers video images to a TMC and has a requirement to detect traffic incidents automatically. The architecture could rely on older, proven technology, such as sending live analog video images along fiber optic cables to the TMC. This architecture is shown in Figure 3.8 and represents the Houston TranStar system in 2002.

Figure 3.8 – Houston TranStar Communication Backbone Architecture
(Circa 2002)



Video images are routed on single-mode fiber optic cables from cameras to hubs where the signals are multiplexed into a single cable and routed to satellite centers. There, video switches route selected multiplexed video to the TranStar TMC, where another video switch is used to further select the desired camera's video. For an operator to select a camera to view its video, control of hub switches and satellite switches is needed. This design originated in the 1980s. It relies on analog technology that is readily available and relatively low in cost, but reliable. The downside is that it is costly to add additional cameras and sensors to the network.

However, digital technology offers a different architecture that has both benefits and risks. An example would be the Gigabit Ethernet communication backbone designed for the Texas Department of Transportation (TxDOT) in Corpus Christi, Texas. Although this network is not built yet, the Miami-Dade Expressway Authority (MDX) is planning to implement this architecture using digital technology. Figure 3.9 is a high-level system functional block diagram of the proposed Corpus Christi advanced traffic management system (ATMS) that uses digital technology. The benefit is that real-time, full-motion video can be shared among five stakeholder groups as well as control of dynamic message signs (DMS) and traffic sensor data. The downside is that at the time of the design (i.e., in the summer of 2002), Gigabit Ethernet switches and routers had not yet proven reliable in field conditions. Since then, field-tested devices have starting appearing on the market.

An example of how a cost-to-benefit analysis is done is provided using these two examples. A baseline number of cameras, DMSs, and their locations were established for the Corpus Christi area. Next, equipment costs were obtained from catalogs and vendors, and the cost data available from similar projects. For analysis purposes, a third architecture was developed using asynchronous transport mode (ATM) technology. Each architecture baseline was priced using the data sources mentioned and the cost results of each were estimated as shown in Tables 3.1, 3.2, and 3.3.

Table 3.1 – Asynchronous Transport Mode Cost Estimates

ATM Device	Quantity	Approximate Cost
MPEG2 Encoder for Each Camera	40	$176,000
MPEG2 Decoder at Each Monitor Station	10	$27,000
ATM Switch	1	$12,000
Chassis and Power Supply	1	$1,400
Patch Panel	1	$1,000
Switching Software	1	$6,500
ATM Multiplexer[a]	5	$80,800
	Total	**$304,400**

[a] Prices are discounted for government purchase.

Table 3.2 – Analog Backbone (e.g., Houston TranStar)

Equipment	Quantity	Unit Cost	Total Cost	Description	Source
Satellite / TMC (1)					
Fiber Distribution Unit (Center?)	1	$400	$400		PBS&J Cost Database
Fiber Optic Receiver	1	$1,200	$1,200		PBS&J Cost Database
4" x 16" Video Switch	1	$3,000	$3,000		Cornet
Video Multiplexer	1	$7,000	$7,000	$11,000 with a demultiplexer – 8 video ports	Cornet
Radio Frequency (RF) Combiner	1	$2,500	$2,500		Cornet (not a Cornet product)
RF to Fiber Optic Converter	1	$6,000	$6,000		Cornet (not a Cornet product)
Video Control Unit	1	$2,500	$2,500		Cornet
Drop and Insert (D/I) Serial Multiplexer	2		$0		
Hubs (5)					
Fiber Optic Transceiver	40	$1,200	$48,000		PBS&J Cost Database
Limited Distance Modem	45	$375	$16,875	Stand-Alone	Traf-Tex, Inc.
D/I Serial Multiplexer	5				
Equipment Cabinets (40)					
Lightning Protection Circuit	120	$50	$6,000		Electronic Specialists, Inc.
Ground Loop Isolator	80	$100	$8,000	Composite Video Isolator	Jensen Transformers
Limited Distance Model	40	$375	$15,000	Stand-Alone	Traf-Tex, Inc.
Manchester Code Converter	40		$0		
Manchester PTZ Controller	40	$399	$15,960	American Dynamics' product	Detection Dynamics
		TOTAL	**$132,435**		

Table 3.3 – Gigabit Ethernet Backbone

ATM Device	Quantity	Approximate Cost
MPEG2 Encoder for Each Camera	40	$80,000
MPEG2 Decoder for Each Monitor	10	$20,000
Fast Ethernet Switch or Hub	40	$80,000
Gig-Edge Switch[a] (Layer 2 Switch)	5	$13,805
Multicast Router[a] (Layer 3 Switch)	1	$11,423
Terminal Server / Intersection	40	$20,000
	Total	**$213,228**

[a] Prices are catalog prices and are not discounted.

Figure 3.9 – Example of a Digital Communication Backbone Architecture

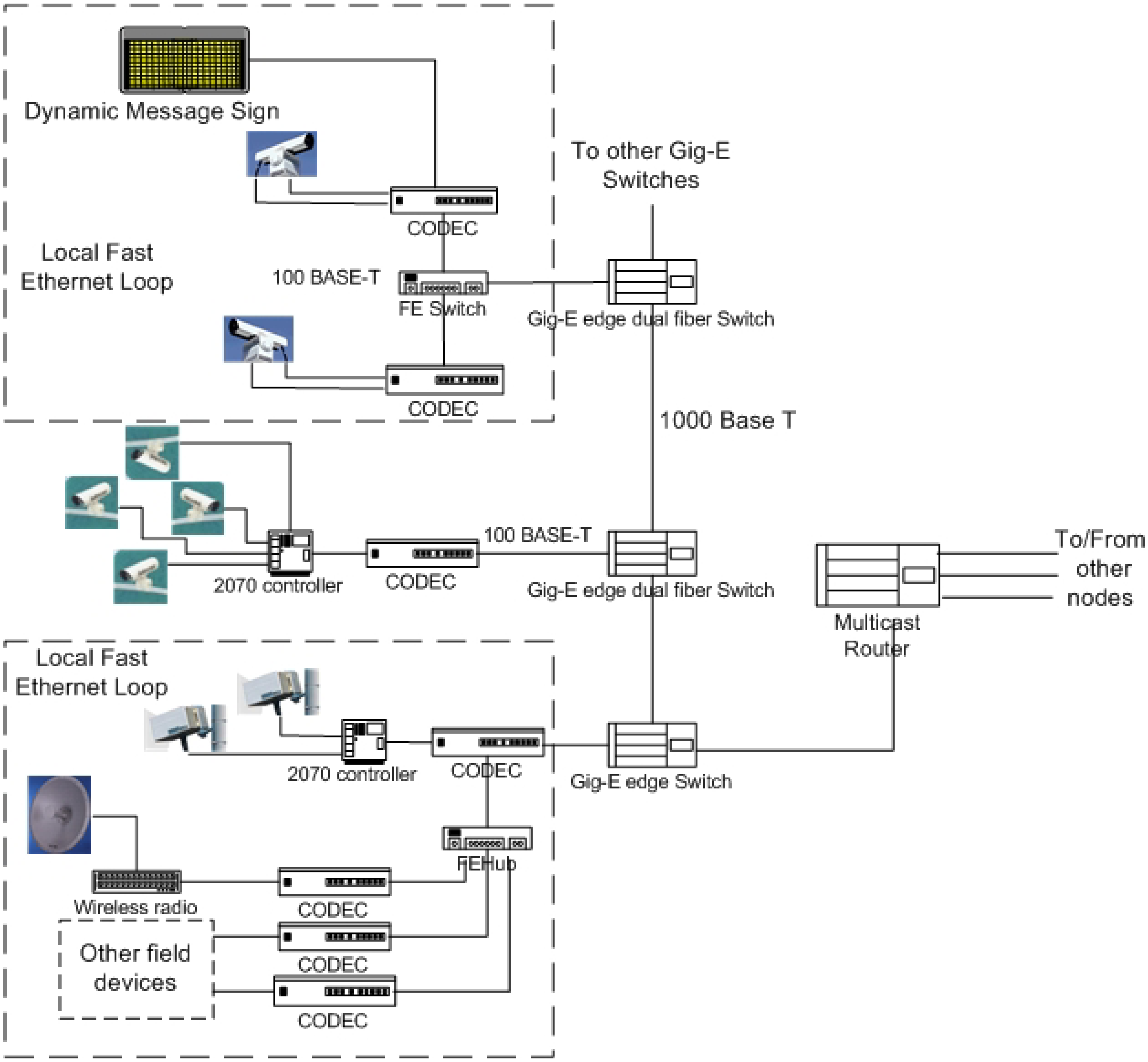
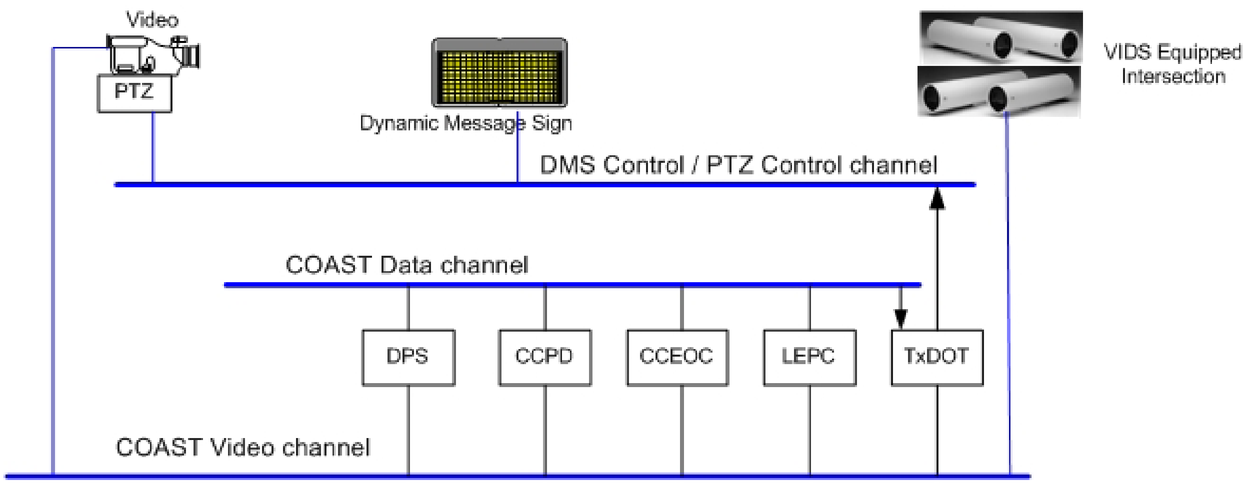


Figure 3.10 shows how the digital communications backbone would be integrated in the system architecture.

Figure 3.10 – How the Digital Communication Backbone Fits the Project Architecture



The cost estimates are called ROM estimates and should actually be called a price since they include an estimate of all markups and taxes. For the three technologies analyzed, Table 3.4 summarizes the ROM costs.

Table 3.4 – Rough Order of Magnitude Cost Comparison

Technology	ROM Price
ATM Backbone	$ 304,400
Analog Backbone	$152, 535
Gigabit Ethernet	$ 213,228

From the comparison, it is clear that the older technology used by TranStar is the cheapest, yet a benefits analysis is also needed to see if each technology satisfies the requirements for full motion video and control among several agencies. The benefits are shown in Table 3.5.

Table 3.5 – Benefits Analysis

Technology	Full-Motion Video to All	Proven Field Equipment	Easy to Add Nodes	Number of Remote Sites Needed
ATM Backbone	Yes	Yes	Yes	None
Analog Backbone	No switch selected	Yes	No costly trenching	Many hubs and satellites
Gigabit Ethernet	Yes	Not yet	Yes	None

From the benefits analysis, it is clear that the analog backbone and the Gigabit Ethernet do not satisfy all the requirements, but the ATM technologies do. However, by the time the system is deployed in 2005, there will be many field-tested Gigabit Ethernet devices on the market. The cost of Gigabit Ethernet is less than ATM, so a rational selection for the ITS project architecture would be the Gigabit Ethernet backbone. As part of the SEP, the ITS project engineer would recognize the risk associated with the Gigabit Ethernet network and the risk mitigation plan (see *Section 3.2*) would have identified the risk that the Gigabit Ethernet field devices are unreliable in hot, wet weather. A risk mitigation strategy would be to use the ATM technology. The decision point of whether to buy Gigabit Ethernet devices or ATM hardware would be identified on the project's critical path network in the master project schedule. (Refer to *Section 4.3.1.7.2*.)

Functional system requirements can be developed independently of the technology to the point where performance or interface requirements dictate that a commitment be made to a particular technology.

Alternate design and technology analysis can be incorporated into the ConOps, or can be a separate document, depending on the complexity and size of the analysis. The example given above combined technology and system architecture, since the technology enabled a different architecture. In some cases, the ITS project engineer will consider alternate designs at a macro level (i.e., the high level) and consider alternate technologies on a low level (i.e., the micro level) within each design. If the technology alternatives will not significantly affect the project design, then the technology cost-to-benefit analysis can be separate from the project design alternatives analysis.

Another factor to consider is legacy systems that the system will interface with, and the constraints imposed on the technology or the architecture. In some cases, alternatives are very limited unless replacing the legacy system is an option.

3.3.7 Procurement Options

Once an architecture and the technology options have been optimized, the ITS project engineer should consider ways to procure and deploy the system. This process is called a trade study in the SEP vernacular because trade-offs are an essential part of the analysis. When doing trade-off studies, the ITS project engineer must consider the system's entire life cycle, especially cost, maintenance, ease of upgrade, and technology obsolescence.

3.3.7.1 Make-Buy Analysis

One procurement option addressed by a trade study is the "make-buy" analysis because the procuring enterprise may have some ability to build parts of the system. If that's the case, can the enterprise do it more cheaply with less risk? If the requirements are well defined by following the SEP, it will be less risky to have a contractor build a product than for an enterprise to do it themselves. A contractor experienced in building similar products who has a good business history can be held accountable through good contracting practices. Other times, when it is very hard to exactly specify the interfaces or requirements, it will be cheaper for the enterprise to build the product *if they have the in-house expertise to do so*.[35]

3.3.7.2 Gap Analysis

A gap analysis is a type of trade study that evaluates an existing system's capability to satisfy the needs of the system being designed. It is called a "gap analysis" because it determines the gap between the current system's capabilities and the new system's required capabilities. Before a gap analysis can be performed, the system functional requirements must be determined based on stakeholder needs. A gap analysis is a lot easier and takes less time if the existing system has a well-documented set of requirements that it satisfied through an acceptance test process. Without that documentation, the engineering team needs to evaluate the existing system's capabilities and "reverse engineer" the requirements to form a basis for analysis. The gap analysis results will affect the procurement options because it may turn out that parts of the legacy system should be retained and the original vendor of the equipment be contracted with to perform selected upgrades. Another result could be that the "gap" is too great between what the system can do and what is required, and the legacy system should be replaced with new technology or new functionality.

Types of procurement contracts are discussed in detail in *Chapter 4* and will not be discussed in this section.

[35] The author places emphasis on this point.

3.3.8 Systems Procurement Documentation

The SEP front-end process results in a set of system requirements that can be used to procure the system. Stakeholders are involved in the SEP from the start and, through formal reviews, have a chance to confirm that the requirements accurately address their system performance expectations. *Chapter 4* of this document discusses program/project planning and addresses the system management process in detail. That process will only be mentioned briefly here.

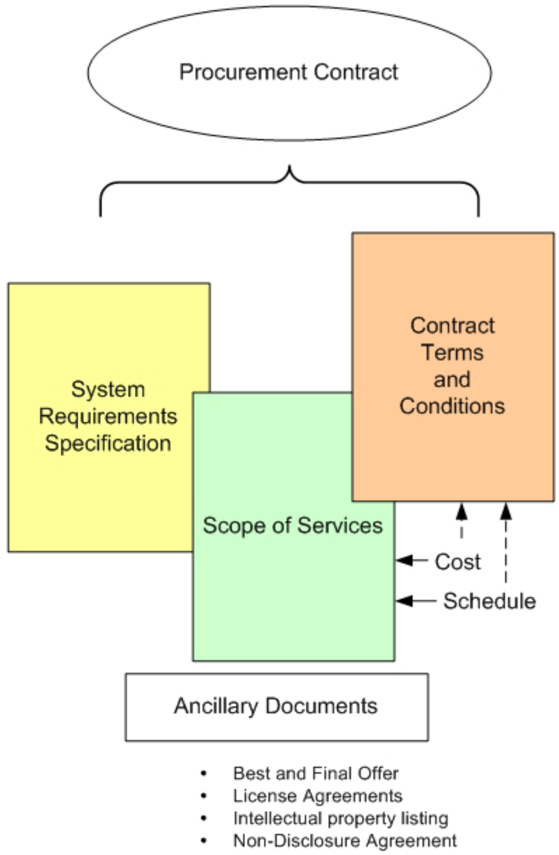
The contractual documentation used to procure a system consists of three basic documents defined below and shown in Figure 3.10.

- The system requirements specification follows a standard format that specifies the system's technical requirements. Refer to *Appendix G* for a template and document development guide.

- The scope of services follows a standard format to specify the project's work requirements, not technical requirements, and is described in more detail in *Appendix P*, where a template and document development guidelines can be found.

- The standard FDOT contractual terms and conditions document is typically provided by the FDOT Contractual Services Office (CSO), and its format and content depend on the type of procurement contract used for the project.

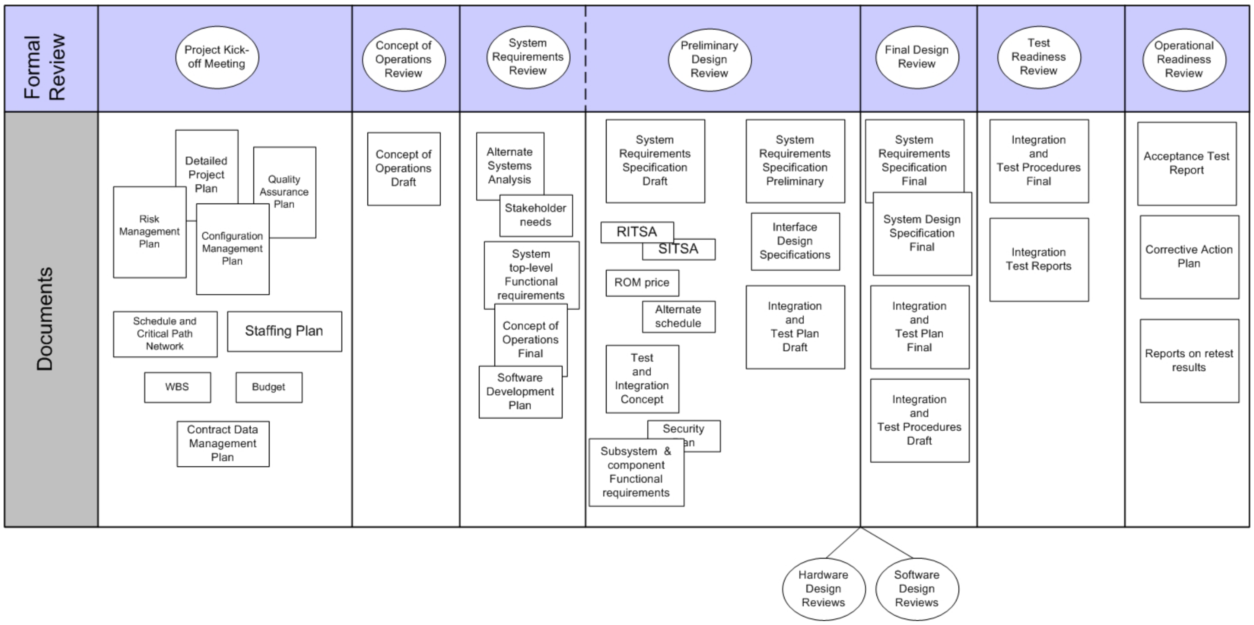
Figure 3.11 – Procurement Documents



3.3.9 Brief Description of the Project Formal Reviews

The formal and informal project reviews and meetings are discussed in detail in the *Section 4.3.1* in the context of project management. Briefly, the documents required to support an ITS project according to the SEP are shown in Figure 3.12.

Figure 3.12 – Documents Used in Systems Engineering Process Reviews



Not all documents are needed, nor are all reviews required by the SEP. The number and level of detail depend on the project size, complexity, and stakeholders. *Section 8* of this document provides guidelines to tailor the SEP to meet project-specific needs. The dotted line indicates that the SRR may be combined with the SDR. The ITS engineering team shall conduct reviews, including design reviews (i.e., system, subsystem, component, life-cycle processes, test readiness, and production approval) and audits (i.e., functional and design configuration), for the purpose of assessing technical progress. Audits typically involve verifying that the documentation matches the product being developed. Normally, a design review should be conducted at the completion of each phase of the project's master schedule. Each review should include:

- An assessment of system requirements and allocations to ensure that requirements are unambiguous, consistent, complete, feasible, verifiable, and traceable to top-level system requirements
- An assessment of design maturity based on technical development goals; master schedule events and accomplishments; and empirical analysis and test data supporting progress to date
- A presentation of the risks associated with a continued development effort

- An assessment of life-cycle processes and infrastructure necessary for product sustainment throughout the system's life cycle
- Verification that resources required for continued development are available
- A determination of whether to proceed with the next phase of the SEP, to discontinue development, or to take corrective actions before proceeding with the development effort

Component and subsystem reviews, and system design reviews (SDRs) shall be conducted, as appropriate, for each level of development. Depending on the system's complexity, low-level reviews may be needed. Often in software developments, the SEP is applied to each software release with multiple design reviews. Trade-off analysis and verification results should be available during design reviews in order to substantiate design decisions. Reviews may result in the need to iterate through the SEP to resolve identified discrepancies before proceeding further in the development activity. Component, subsystem, and system functional and design configuration audits shall be performed to ensure that: supporting documentation has been satisfactorily completed; qualification tests for each specification requirement have been completed and all requirements satisfied; and/or that products comply with final drawings.

3.3.10 Functional Verification – The Back-End Systems Engineering Process

The SEP shall include a functional verification process. Its purpose is to assess the completeness of the functional architecture or detailed design in satisfying the validated requirements baseline, and to produce a verified functional architecture that establishes the system problems to be solved by the project. This process verifies that the functional architecture or detailed design defined by the ITS engineer is upward traceable to user expectations and constraints. In addition, all user expectations and constraints need to be downward traceable to the functional architecture and detailed design requirements.

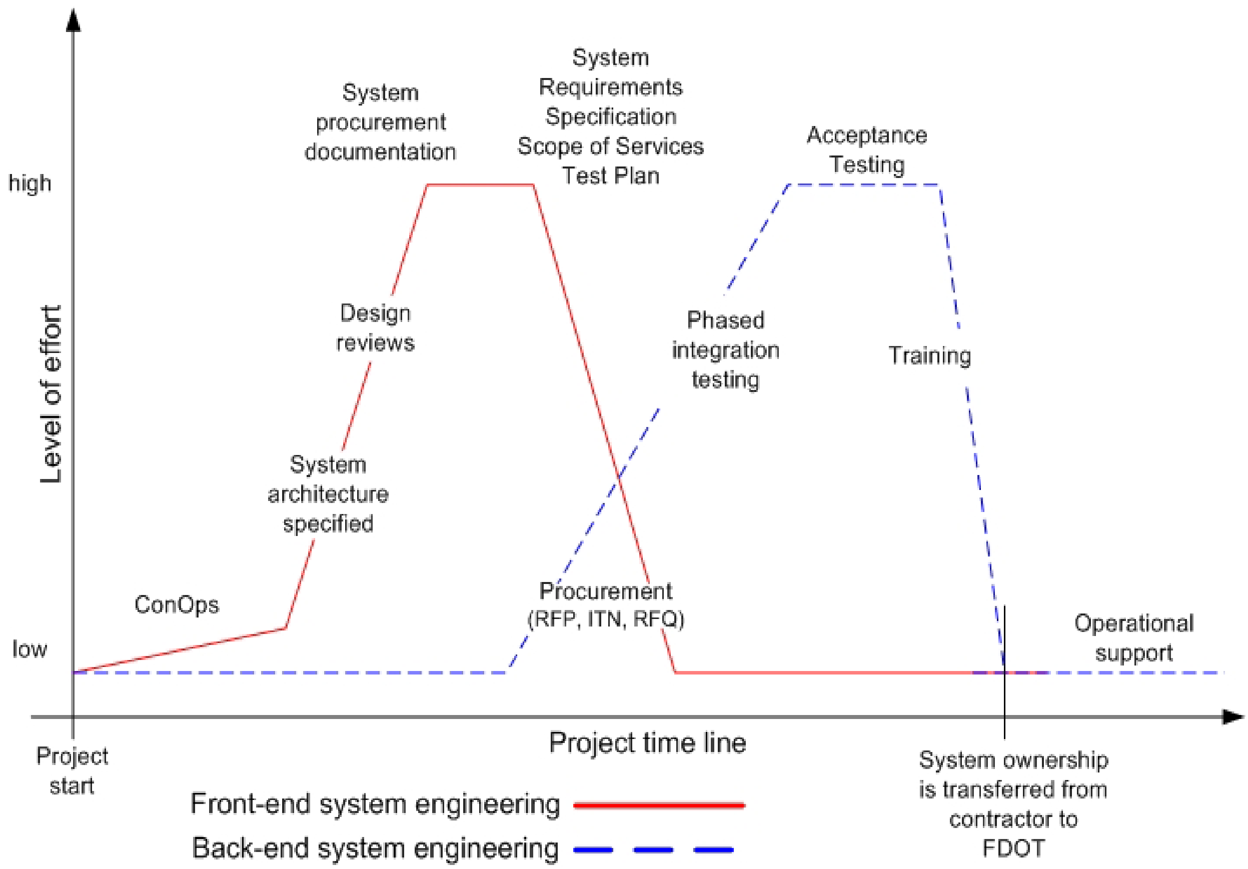
The requirements obtained from the *SITSA* and *NITSA* need to be traced to stakeholder requirements and constraints. The database tool discussed in *Section 4.1.3.3* that will be used to map these constraints, architecture functions, and equipment packages helps to ensure all parent-child relationships trace back to stakeholder requirements (i.e., user expectations and constraints). Based on this traceability, the identified functions should to be modified, deleted, or added, as necessary. More explanation of how to use the database requirements tracking tool is provided later.

The back-end process completes the development cycle of the project and transitions to the sustaining phase. A good test program that uses phased integration along functional threads will specify the conditions that indicate when system development will be complete and the system ready for customer acceptance. Along the way, the test program not only verifies that the requirements have been met, but also characterizes the system's operation, identifies its limits, and helps finalize the operations and maintenance concept.

The test phase of the SEP actually begins soon after the project has started and requirements have been defined. It is strongly recommended that a test director be identified and staffed before project kick-off. The test director's focus will be to ensure that the identified requirements are specified in such a way that they are testable. As requirements are being specified and allocated to the system architecture, the test director is already thinking of test scenarios that can prove that a requirement or group of requirements has been satisfied by the system.

There are two distinct phases of the SEP: the front-end phase that establishes the requirements, the system architecture and, in some cases, the detailed design, and the back-end phase that proves that the delivered system meets all the requirements. The level of effort associated with each phase can be graphed against the project development time line. As the front-end systems engineering effort plateaus and declines, the back-end systems engineering effort rises to a climax at the system acceptance test, dropping off afterwards to the sustaining level. This is graphically shown in Figure 3.13.

Figure 3.13 – Effort Distribution Level Between the Front-End and Back-End Systems Engineering Efforts



3.3.10.1 Classification of Test Methods

Each requirement will be tested using one of the following four test methods:

- Inspection (I)
- Analysis (A)
- Demonstration (D)
- Test (T)

3.3.10.1.1 Inspection

The acceptance test procedures are followed in the review and/or inspection of the end item, including its drawings and characteristics. As an example, review and/or inspection could be used to verify that the following requirement is met:

> CH001L – The transaction report file shall be a comma-separated, value-formatted file that summarizes transactions according to date, block, and route, so that all transactions for a particular block and a particular route are summarized for the date collected.

This requirement would be verified by obtaining a transaction report from the system and examining it to verify that it is a comma-separated, value-formatted file and that the data is summarized as required.

3.3.10.1.2 Analysis

Analysis is a verification element in the form of a statistical study of previously collected data. It results in calculated data that is intended to verify a requirement when an examination, test, or demonstration cannot feasibly be used to verify the requirement. Such data, collected during a tightly controlled test setup, may be composed of a compilation of acceptance test data, design solutions, or data derived from low-level tests. Satisfaction of the requirement is performed by statistical analysis of the test data. As an example, analysis could be used to verify that the following requirement is met:

> CH015 –The clearinghouse subsystem shall generate needed reports based on transaction logs resulting from card and device activity.

This requirement could be verified by running transactions under controlled conditions, keeping track of the amounts that should have been transacted. The test director would then generate the reports from the system and verify that the transaction numbers are what were expected. This would involve an analysis of the transaction details as reported by the system.

3.3.10.1.3 Demonstration

Demonstration is an element of verification that differs from testing in that it verifies only the specific situation demonstrated but not the total requirement. Demonstration is used in lieu of testing where system parameters cannot be sufficiently controlled to provide a test that verifies the stated requirement explicitly. In such cases, performance within the stated requirements will be demonstrated for the specific case or cases. The capability to conform to the requirement must be inferred from the successful completion of the specific demonstration. The bulk of system testing should be demonstration tests because they are relatively easy to set up and execute. The data requirements for a demonstration test are minimal compared to a test-type method. An example of a demonstration testing method is given by the following requirement:

> CH002L – The transaction summary file may be in any format that can be opened by Microsoft® Word® or Excel® (Version 95 or later).[36]

This requirement could be verified by taking a summary transaction file and opening it using the Microsoft Excel program, and then opening it using Microsoft Word. The requirement only stipulates that either Excel or Word be able to open the file; therefore, by testing both programs, the test director could characterize the system's ability to perform. If the file couldn't be read using Word but could using Excel, or vice versa, the requirement would have been met and the software would have passed the test. Taking the test a step further by using both programs produced more information about how the system works, which helps in the support and maintenance phase of the SEP.

3.3.10.1.4 Testing

Testing is an element of verification that denotes the determination of the properties and characteristics of equipment or components by technical means, including functional operation, and the application of established test principles and procedures. The analysis of data derived from a test is an integral part of this verification element and should not be confused with analysis. An example of this verification element is given by the following requirement:

> LR009 – The maximum balance that the load/reload transaction subsystem shall permit in the e-Cash purse shall be $99.

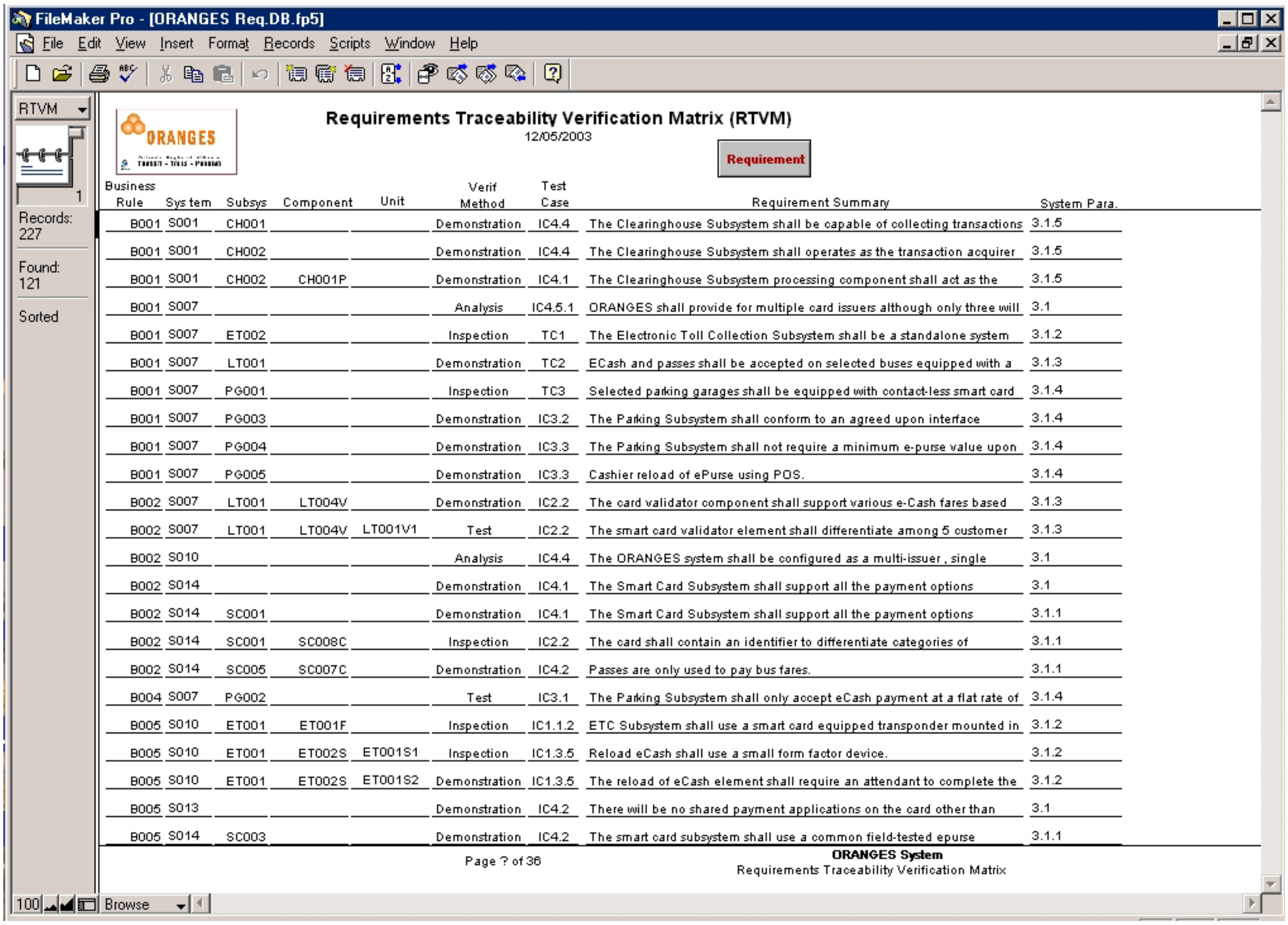
[36] Microsoft, Word, and Excel are registered trademarks of Microsoft Corporation in the United States and/or other countries.

This requirement is a test because a numerical value can be measured and tested against. This requirement can be tested by entering a transaction of $98.99, then seeing if the system accepts it and loads that amount on the smart card. Next, the transaction amount would be one cent and the system is still expected to accept it, since the e-Cash purse value limit is $99. Finally, another transaction of one cent would be made, with the system being expected to reject the transaction since the e-Cash purse value would exceed $99 if the transaction were accepted.

3.3.10.2 The Requirements Traceability Verification Matrix – An Evolution of the Requirements Traceability Matrix

The requirements traceability verification matrix (RTVM) is the way the test director keeps track of where each requirement is being tested and how it is being tested. This table is an outgrowth of the RTM mentioned in *Section 4.1.3.3*. The RTVM should be started as soon as the system level requirements have been defined and an RTM is created by the ITS project engineer. A sample RTVM is shown in Figure 3.13. The RTVM example is from a smart card project in Orlando, so business rules take the place of stakeholder requirements (i.e., needs). The database is sorted in the system architecture's hierarchal order, so that a system requirement is the parent of one or more subsystem requirements, and so on. Some system-level requirements do not have child requirements because the requirement is stated unambiguously for the entire system. The verification test method is listed, as is the test case (TC) where the requirement is tested. The system specification paragraph that contains the paragraph is also listed in the database of the requirements. The letters "IC" in the Test Case column refer to integration cases (ICs) that are an integral part of a phased I&T process described in the following sections.

Figure 3.14 – Sample Requirements Traceability Verification Matrix



FileMaker Pro - [ORANGES Req.DB.fp5]

File Edit View Insert Format Records Scripts Window Help

RTVM

Records: 227

Found: 121

Sorted

Requirements Traceability Verification Matrix (RTVM)

12/05/2003

Requirement

Business Rule	System	Subsys	Component	Unit	Verif Method	Test Case	Requirement Summary	System Para.
B001	S001	CH001			Demonstration	IC4.4	The Clearinghouse Subsystem shall be capable of collecting transactions	3.1.5
B001	S001	CH002			Demonstration	IC4.4	The Clearinghouse Subsystem shall operates as the transaction acquirer	3.1.5
B001	S001	CH002	CH001P		Demonstration	IC4.1	The Clearinghouse Subsystem processing component shall act as the	3.1.5
B001	S007				Analysis	IC4.5.1	ORANGES shall provide for multiple card issuers although only three will	3.1
B001	S007	ET002			Inspection	TC1	The Electronic Toll Collection Subsystem shall be a standalone system	3.1.2
B001	S007	LT001			Demonstration	TC2	ECash and passes shall be accepted on selected buses equipped with a	3.1.3
B001	S007	PG001			Inspection	TC3	Selected parking garages shall be equipped with contact-less smart card	3.1.4
B001	S007	PG003			Demonstration	IC3.2	The Parking Subsystem shall conform to an agreed upon interface	3.1.4
B001	S007	PG004			Demonstration	IC3.3	The Parking Subsystem shall not require a minimum e-purse value upon	3.1.4
B001	S007	PG005			Demonstration	IC3.3	Cashier reload of ePurse using POS.	3.1.4
B002	S007	LT001	LT004V		Demonstration	IC2.2	The card validator component shall support various e-Cash fares based	3.1.3
B002	S007	LT001	LT004V	LT001V1	Test	IC2.2	The smart card validator element shall differentiate among 5 customer	3.1.3
B002	S010				Analysis	IC4.4	The ORANGES system shall be configured as a multi-issuer , single	3.1
B002	S014				Demonstration	IC4.1	The Smart Card Subsystem shall support all the payment options	3.1
B002	S014	SC001			Demonstration	IC4.1	The Smart Card Subsystem shall support all the payment options	3.1.1
B002	S014	SC001	SC008C		Inspection	IC2.2	The card shall contain an identifier to differentiate categories of	3.1.1
B002	S014	SC005	SC007C		Demonstration	IC4.2	Passes are only used to pay bus fares.	3.1.1
B004	S007	PG002			Test	IC3.1	The Parking Subsystem shall only accept eCash payment at a flat rate of	3.1.4
B005	S010	ET001	ET001F		Inspection	IC1.1.2	ETC Subsystem shall use a smart card equipped transponder mounted in	3.1.2
B005	S010	ET001	ET002S	ET001S1	Inspection	IC1.3.5	Reload eCash shall use a small form factor device.	3.1.2
B005	S010	ET001	ET002S	ET001S2	Demonstration	IC1.3.5	The reload of eCash element shall require an attendant to complete the	3.1.2
B005	S013				Demonstration	IC4.2	There will be no shared payment applications on the card other than	3.1
B005	S014	SC003			Demonstration	IC4.2	The smart card subsystem shall use a common field-tested epurse	3.1.1

Page ? of 36

ORANGES System
Requirements Traceability Verification Matrix

100 Browse

3.3.10.3 Integration and Testing Planning

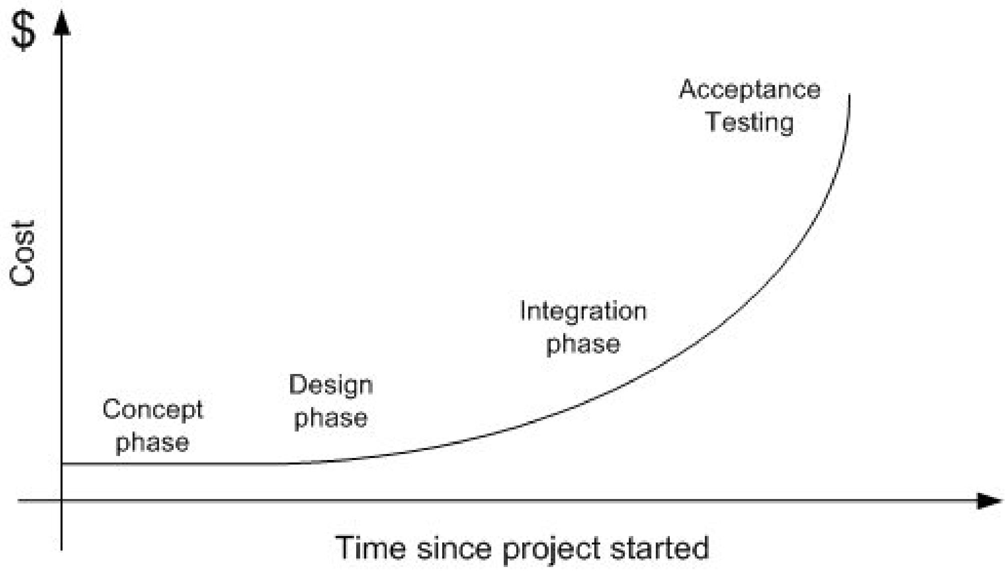
Planning for system I&T is as important as the specification of the system architecture and requirements. Whether a contractor has responsibility for building the system and "selling it off" to the FDOT, or the FDOT is responsible for pulling together products from different contractors and integrating them into a system, the I&T process remains the same but is applied at different levels. For the purpose of the *SEMP*, it is assumed that the FDOT has contracted for a company to deliver a working system of hardware and software that meets the RITSA requirements defined by the front-end SEP. Ideally, the FDOT should have someone other than the contractor developing the system write the acceptance test plan and procedures. This is called the independent verification and validation, or IV&V, process, which was previously introduced. Further, the ITS project engineer should insist that the system be integrated along functional threads with demonstrated milestones that can be tied to progress payments.

3.3.10.3.1 Phased Integration with Demonstrated Milestones

It is true that a company wants to control a project to minimize change; deliver a product on time and within budget; and to maximize their profit. Companies tend to view phased integration with demonstrated milestones as a risk that can lead to requirement changes, schedule slippage, and cost overruns. This is generally because most companies have not worked with a customer who had a standardized SEP and well-documented, unambiguous requirements. The mutual agreement on requirements and a well-planned I&T plan alleviates contractor risk and even offers the opportunity for managed change that can result in higher fees.

The benefit to the FDOT is that the process manages stakeholder expectations and eliminates surprises. For this process to work well, the stakeholders and the contractor must be committed to it so that a series of small problems are uncovered and fixed during development, instead of a large problem that is uncovered during acceptance testing and is expensive to fix. Generally, the cost to fix a problem in a system increases as the square of the time since the project started. This can be easily understood by using the example of finding an error in the interpretation of a requirement that is discovered while reviewing the ConOps. The cost to correct the problem is to change a few pages in a document, restate a few requirements in the database, and adjust the architecture accordingly. But if an error is found in the interpretation of a requirement after the system has been integrated and is being tested for acceptance, the change involves redesign, recoding, unit testing, integration testing, and revision of previously published documentation. This concept is shown in Figure 3.15, and underscores the importance of getting as much visibility into the system during the development process as possible and verifying progress through milestone demonstrations that produce tangible, verifiable results.

Figure 3.15 – Cost to Fix Problems Increases Exponentially During the Project Life Cycle

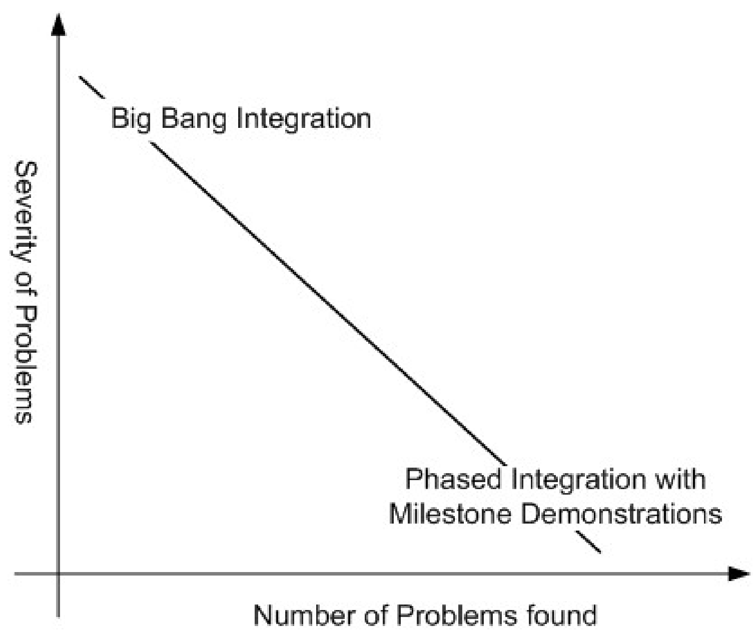


Phased integration with demonstrated milestones is generally more expensive for the contractor and involves more work for everyone. If the FDOT is too invasive during product development, the contractor's cost will increase and the schedule will slip, yet if the FDOT is not a partner in the system development, stakeholder expectations will most likely not be met. It is a fine line to walk between too much and too little involvement. A lot depends on how much mutual trust exists between the FDOT and the contractor. Also, small, short-term projects do not require much phased integration or milestone demonstrations, whereas large, complex projects require it.

Companies generally do not want to have the FDOT involved in development testing, preferring instead to deliver the system and conduct one acceptance test at the end of the project. This is called the "big-bang" integration approach and happens when a company doesn't test against requirements during factory integration. Rather, the company puts all the pieces of the system together and then tests it during acceptance testing. Big-bang integration typically happens when the requirement specifications and scope of work are loosely defined. It is called "big bang" because typically that is what happens when the system is turned on – nothing works right.

The phased I&T process uncovers small problems early in the development cycle of the system, whereas the big-bang approach results in little information about the technical progress of the project until the end. Too little information is often interpreted as a sign of good progress and schedule adherence throughout most of the project's development cycle until acceptance, where problems are glaringly obvious. The problems may be few at that point, but are usually catastrophic, so the schedule and budget are severely impacted and management has few, if any, options left except to spend more money to fix the problems. This is graphically illustrated in Figure 3.16.

Figure 3.16 – Integration Approach Characteristics



It is perceived by some contractors that phased integration is more costly because of the formality involved, such as the need to manage the system configuration formally, and the perceived need for early roll out of the training and documentation packages. (Refer to *Section 6.1.4*.) In reality, the costs saved by phased integration greatly outweigh the big-bang approach on medium to large projects. Further, milestone demonstrations need not require the formality associated with final product installation and acceptance.

3.3.10.3.2 Milestone Demonstrations Tied to Progress Payments

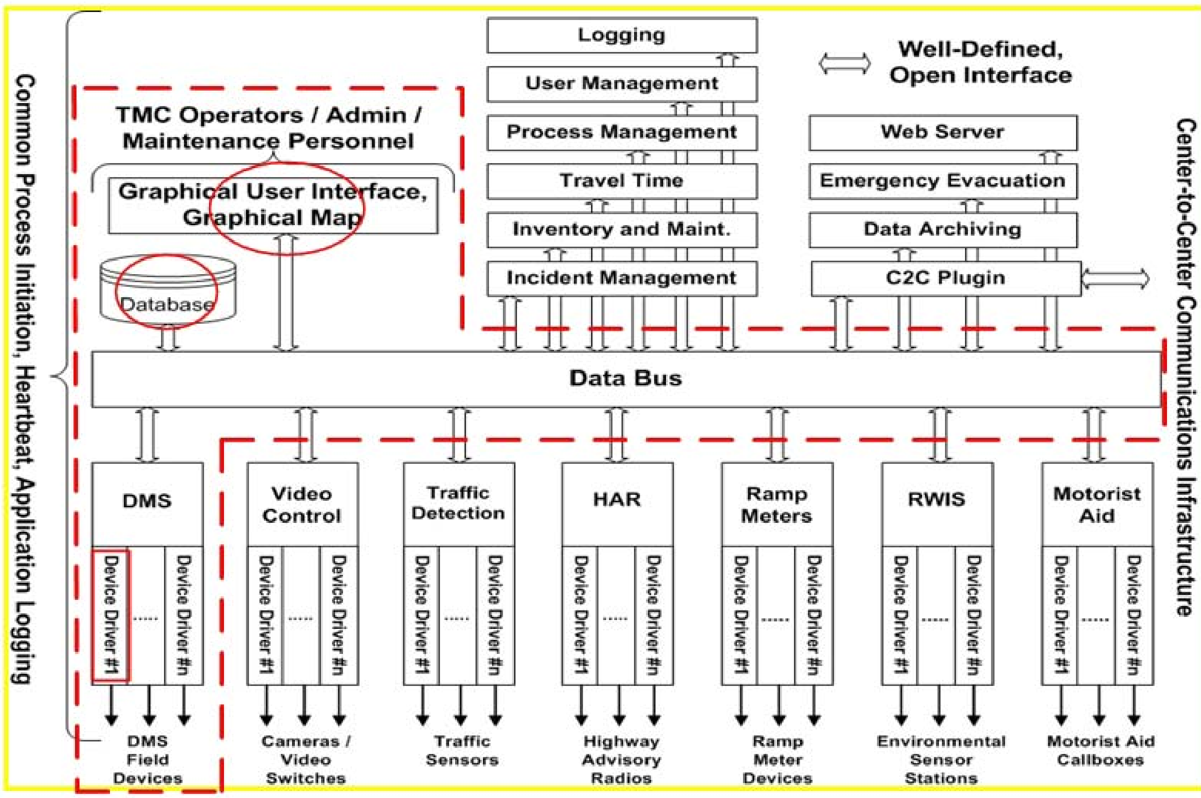
The concept of identifying key system functions, and having them assembled and demonstrated at the customer's site is new to the ITS market. Most contractors would probably prefer not to have to tie payments to verifiable performance, but it is about the only way the FDOT can really measure the progress of an ITS project that has a lot of technical development, such as projects with a lot of software development. During precontract negotiations, the FDOT and the selected contractor should work to identify key functionality that can be readily demonstrated on-site, then negotiate the monetary value of each milestone demonstration. Ideally, the phased integration of the system builds on a foundation that allows system users to continue to operate the partial system after the milestone demonstration. This affords them more opportunity to uncover system faults before the final system is fixed, and possibly provides opportunities for contract engineering change requests (ECRs) that will result in a better system for the FDOT and an additional fee for the contractor. It should be recognized by all parties that if the partial system remains in place after the milestone demonstration, the contractor's support task will need to be activated earlier in the schedule than would be normal with a more traditional project plan. Another issue to negotiate is the partial ownership of the system if parts of it are left at the customer's site after a milestone demonstration. Ideally, the system's ownership should not transfer until final system acceptance, so the contractor still owns the equipment and software, and a temporary transfer of custody must take place if the equipment is to be left on-site after a milestone demonstration.

3.3.10.3.3 An Example of Phased Integration with Demonstrated Milestones

The best way to understand the concept of phased integration with demonstrated milestones is to study an example based on an actual FDOT project. The SunGuide Software System[37] is a large software system to be used by the FDOT Districts, Florida's Turnpike Enterprise, and MDX to manage the ITS resources in RTMCs. Figure 3.16 is a functional block diagram of the software system architecture.

[37] More information regarding the FDOT's SunGuide Software System is available online at http://stmcsls.datasys.swri.edu/.

Figure 3.17 – DMS Functional Thread for the SunGuide[SM] Software System



The fundamental building block of the software system is the data bus that serves as a conduit for all messages and data throughout the system. When a piece of software attached to the data bus wants to send a message to another piece of software to control a device or retrieve information from a database, the software places a message in a predefined format on the data bus, which has the intelligence to deliver the message and data to the intended destination. The SunGuide software system cannot operate without the data bus. This software piece is the fundamental building block of the system.

Another important piece is the user interface to the software. Without the ability for users to interact with the software, there would be no command and control ability for the RTMC. The user interface is typically called a graphical user interface (GUI). The development of GUIs is a little different from the classical waterfall systems engineering design process because it is so subjective. Everybody who will use the system will have an opinion on how it should look and feel.

The best way to develop a GUI is to use the spiral software development method early in the project. Essentially, the spiral method uses rapid prototyping to develop a simulation of the GUI based on perceived stakeholder needs and any system requirements that are available. The GUI operates in a stand-alone fashion, with software simulating the functions of the system when various buttons are pushed and menu options selected. Sometimes, if a CCTV image is to be simulated, a digital recording of a live CCTV image may be used in a loop mode. The idea is to quickly piece together a working model of the GUI and show it to the stakeholders for comments. Feedback from the demonstrations is used to rebuild the GUI in the next evolution. This process repeats several times, ideally converging to a final version, hence the name the "spiral" development method. For more information on the spiral development method, refer to *Section 2.3.7, The Spiral Model,* of *Technical Memorandum No. 1* of this project.

Assume that an FDOT District wants to use the SunGuide software system to operate an RTMC, and they already have 5 CCTV cameras and 10 DMS units installed in the field. However, the signs are controlled by stand-alone software using a dial-up connection, and the CCTV cameras have their video individually sent to the control room through individual fiber optic cables. Further, assume that the District wants to show early progress to its community, and can't wait for the SunGuide software system to be fully developed and installed in two years. The ITS project engineer, working with the District engineer and the contractor, identifies the ability to control their DMSs as a high priority function that is needed immediately. A secondary priority is to have the CCTV images integrated into a video wall through a central control tied to the DMSs.

This would be accomplished by a phased integration of the software and hardware on-site with demonstration milestones. Figure 3.16 has a red, dashed line drawn around the parts of the SunGuide software system that would be needed for the first milestone demonstration. This is called functional thread testing because all the pieces of the system that are needed to demonstrate the DMS command and control function are integrated, so a functional thread flows through the system. The parts of the GUI needed to view and control the DMSs include the parts of the database needed to support DMS messages, the data bus, the DMS driver, and the data communication link to the DMS. The ITS project's test director will develop ICs that will be used to integrate the SunGuide software system pieces on-site as the contractor completes them. The ICs culminate in a milestone demonstration of the ability to place messages on a selected DMS in the field. Various system requirements can be tested and verified during IC testing that can serve as a foundation for later acceptance testing.[38] The demonstration milestone could be linked to a progress payment to the contractor upon the successful demonstration of the DMS function. The second phase of integration would add the GUI parts that support CCTV, the database, and driver portions so CCTV images could be controlled by an operator at a SunGuide software system workstation. By carefully selecting a CCTV camera near a DMS, the milestone demonstration could show integrated operation by placing a message on the DMS and viewing that message through the CCTV image.

[38] If IC testing can verify that low-level requirements have been met, then acceptance testing can take place at a higher level, resulting in a faster, less expensive acceptance test.

By following this phased approach to system integration, the early parts of the District's system are tested many times, resulting in a much more reliable product. Another advantage is that when a new function is added and the system ceases to work properly, the test director knows it had to be related to the function just added. This makes system troubleshooting easier and faster. If there are several contractors supplying parts to the system and the FDOT is managing the system integration, a phased approach helps identify who is responsible for problems when they arise.

3.3.10.4 Integration and Testing Documentation

The back-end processes have fewer documentation requirements than the front-end processes, but the documentation is used by the test engineers as it is refined and finalized. The basic components of the I&T documentation are the test plan or, more appropriately, the I&T plan, the testing procedures (both ICs and TCs), and test reports. The CM process is a separate but associated procedure used to create test baselines for the system and control changes to those baselines. The requirements database supports the back-end process as it did in the front-end by documenting where in the test procedures the specific requirements are tested and whether it has been verified that the system under test (SUT) meets the requirements. In testing, validation that the requirements are stated correctly and that they describe the required system behavior must be attained.

3.3.10.4.1 Test Plan

Most projects refer to a test plan that describes the system's general test plan for acceptance. In many cases, the test plan infers test procedures that are generally to be followed. The recommended FDOT SEP requires that the ITS project engineer and the development team plan the system's phased I&T. As described in *Section 4.2.3*, phased integration with demonstrated milestones provides the FDOT with real visibility into the system's development, and allows the FDOT to spot problems early and to fix them before they get to be large problems. The I&T plan is started as soon as system requirements are identified, and the requirement allocations and system synthesis process has started. Each requirement needs to be reviewed by the project's test director for testability.[39] The wording of each requirement is adjusted to fit a general idea of how a requirement may be tested. This information needs to be captured in the requirements database and attached to the applicable requirement. Based on the system functional architecture and stakeholder needs, a set of demonstrable functional threads can be identified that run through the system. The I&T plan identifies the functional parts of the system that are needed to support the functionality to be tested and this milestone demonstration is identified as an IC. The system developer must provide a schedule for when the system pieces will be available for integrated testing by the FDOT so the time phasing of the ICs will mesh with the project's master schedule.

[39] The test director's role may be performed by the ITS project engineer, or the project's system engineer on smaller projects, or may be the same person until the PDR, and then the responsibility is transferred to someone dedicated to leading the test effort.

The I&T plan is published as a draft during the preliminary design review (PDR), and as a final during the system's critical design review (CDR) or final design review (FDR). The plan is not a detailed description of testing, but rather presents the overall concept of the I&T plan, and identifies individual ICs and TCs.

The ICs are used to integrate the system and result in demonstrated milestones. Typically, low-level requirements that pertain to the function being demonstrated are tested.

The TCs are used to verify that a system built and released for the FDOT's use meets all applicable requirements. If a system is deployed in stages, the system acceptance test is performed on the final build of the system. The final system acceptance test is referred to as the FAT or SAT, but FAT can also refer to the factory acceptance test. The factory acceptance test is performed by the system manufacturer and takes place at their facilities. If subsystems are being developed by different vendors and the FDOT is integrating them into a single system, the FDOT will require the vendor to prove that its subsystem meets all the system requirements that apply to it. Usually, the FDOT witnesses the factory acceptance test and will often have procedure approval responsibility. The FDOT SEP recommends that the system acceptance test performed by the FDOT on the final system delivery be called the SAT. *Appendices J*, *K*, *and L* are the templates that can be used to develop a standard I&T program for the project. The templates are generic and many of the section headings will depend on the system functional architecture.

1.3.10.4.2 Test Procedures

Test procedures are required by *Florida's Statewide SEMP* because they create a standard way of testing that leads to repeatable results and predictable responses. The general format for test procedures is to state them in tabular form with the step numbered, the test action to be taken described, and the expected result noted. When an expected system response proves or disproves a system requirement, the system requirement identification number is put in the pass/fail column to make it easier to track what requirements have been tested. In general, the test director should design tests that rely on the demonstration test technique, since this and inspection are low-cost methods of testing. Further, if IC tests have already verified low-level requirements, acceptance testing can concentrate on high-level functional tests and include the pervious test results by reference. In practice, IC testing often results in a group of requirements that were not fully satisfied. If the problems are not critical, instead of holding up the system development by further testing, those requirements are bumped to acceptance testing where they must be reconciled. Table 3.6 is a brief sample of actual system test procedures used on an electronic payment system project in Orlando.

Table 3.6 – Test Procedure Sample – IC2.3 Pass Expiration Test

Step	Procedure	Expected Result	Pass/Fail
1	Take card #9 and insert it in the point-of-sale (POS) device and read the pass.	The first pass should read 01 9011 0000 001E 02, indicating that it is a 30-day student pass with auto-renew.	
2	Touch card #9 to the validator.	A beep is heard, the green light emitting diode (LED) lights briefly and the display reads "Pass Accepted."	
3	Take card #9, insert it in the POS device, and read the pass.	The first pass should read 01 9011 XXXX YYYY 02, where XXXX is today's date and YYYY is a number 30 or higher.	
4	While the card is still in the POS device, use the special menu to change the first date to 31 days ago so the last date is yesterday.	The pass should read 01 9011 XXXX YYYY 02, where XXXX is today's date minus 31 days and YYYY is today's date minus 1.	
5	Touch the card to the validator.	A beep is heard and the display shows "Pass Renewed," S020, SC010, CH211.	

Procedures continue for another 15 steps for this particular test. The last column on the right is used to indicate whether each step in the procedure has passed or failed. The procedure is designed so that the procedures can be printed out and used to guide the testing. The test engineer marks "P" or "F" in the last column to record the result of each step. The printed, marked-up copies of the procedure become the test data sheets. See *Appendices J*, *K*, *and L* for templates, and a plan, procedure, and test report development guide.

Writing test procedures is an iterative process, and it typically takes about three passes through the procedures to refine them so they work and the SUT is properly tested. This is because of two things:

- The actual operation of this system is surmised when writing the draft procedures and the actual operation of the system is gleaned from running the procedure.

- There may be actual system failures that need repair before further testing can take place. For example, a sequence of steps the test engineer thought appropriate may, in fact, turn out reversed when tried on the SUT. Or, the expected system response is different from what was anticipated, but upon further analysis is deemed to be correct.

Procedures are never written once and used to test a system. Expect at least three dry runs before running the test officially. What is hard to accept is that even with three dry runs, there may still be system results that do not match the expected results when the test procedure is run officially. This is typically due to a system failure, but may also be a proper system behavior that was not well understood before testing. Post-test analysis will usually uncover these reasons.

Of course, dry runs can continue until the system passes every test, but often the amount of time it would take is prohibitive and would unacceptably delay system deployment. The system can be accepted with bugs.

3.3.10.4.3 Test Reports

A system "bug" is an unexpected result from testing that, after careful analysis, is deemed a wrong system response to a test. In many cases, the response is a minor deviation from the expected result and a work-around is possible. For example, a system is required to automatically start a process when an event is entered in the system database but, for some reason, it doesn't happen. The temporary operational work-around would be for the system operator to click a button that manually forces the event to take place after the data is entered in the database. In this way, the system could be placed in operation, while the vendor fixes the bug and delivers an upgrade to the system. Test reports document test results, and a generic template is presented in *Appendix L*. The test report should quickly summarize test results so the reader can see what happened and what is recommended within the first two pages of the test report. Later sections of the report have the test results, analysis, and recommendations.

3.3.10.4.4 Deviations, Waivers, and Failures

The execution of a test procedure can have one of the following four results:

- **Pass**, where the correct results were observed and the associated requirements were met
- **Failure**, where the correct results were not observed and the associated requirements were not met (i.e., no work-around solutions and the system cannot be used until this bug is fixed)
- **Failure – Deviation**, where the correct results were not observed and the associated requirements were not met; however, there are work-around solutions and the system can be used until the vendor fixes the problem

- **Failure – Waiver**, where the correct results were not observed and the associated requirements were not met; however, there are no work-around solutions, and the cost to fix the system is prohibitive in terms of either time or money. The FDOT accepts the system as-is and grants the contractor a waiver for the requirement. Often, the FDOT would ask for consideration, a contractual term for compensation for accepting a system that is less functional than what was required. It is also possible that after the system was developed, the FDOT and contractor both realize that the interpreted requirement is impractical or no longer applicable. In this case, a waiver can be granted with no intent to ever fix the problem.

Since milestone demonstrations can be tied to progress payments for the contractor, and IC failures are retested in the acceptance test, the failure to ever verify that a requirement has been met can be tied to a contractual monetary amount.

3.3.10.4.5 Formal and Informal Reviews

The amount of documentation required for the back-end SEP is less than the front-end. The recommended SEP requires an I&T planning document, plus test procedure documents, and there are usually many for each major IC or acceptance test. If a large software system is being developed, there may be many SATs tied to software builds. The point of acceptance testing is to formally transfer ownership of the system to the FDOT. In contracts with the United States Department of Defense (USDoD), Form DD250 is a paper used to signify that the ownership of a system is transferred from the contractor to the USDoD.[40] Currently, the FDOT does not have an equivalent form or procedure. If the FDOT partially accepts a system for operation as part of a phased integration and deployment, the system is not accepted by the FDOT until the final SAT. There should be a temporary custody transfer of the hardware and software to the FDOT, so the FDOT is legally responsible for the portions of the system they are operating until the final SAT takes place.

[40] *Form DD250 – Material Inspection and Receiving Report* (August 2000), United States Government Printing Office (USGPO). USDoD forms available online at http://www.dtic.mil/whs/directives/infomgt/forms/ddforms1-499.htm.

3.3.11 Support Phase

Often, the FDOT will contract for system development and then some period of operational support after system acceptance. Since it is nearly impossible to specify exactly what kind of work will be required after the system becomes operational, the contract is usually written for paying support costs in terms of time and materials (T&M). The contractor is paid for the actual hours spent plus overhead, fringe benefits, and a profit, as well as being reimbursed for actual direct costs associated with the support. Typically, there is both a time limit and a spending cap on the T&M. The final SAT is the gateway to transition the contractor from the development phase that may be based on progress payments linked to deliveries or milestones to T&M. Therefore, a smart contractor will realize that a well-defined I&T program with documented test procedures and unambiguous test criteria will also benefit them by specifying when development is completed and the support phase is entered. A contractor can go out of business trying to sell a poorly defined system with constantly changing requirements and following an ill-defined testing process.

Some contractors and some procurement agencies believe that ambiguity and vague requirements provide flexibility in the contract. Often, this is not the case. A project governed by a vague contract can completely fail to achieve its intended goal when management personnel change on either side of the contractual relationship. Always take more time planning and refining requirements before rushing to deploy the system because, ultimately, a structured approach saves both time, money and, often, careers.

3.3.12 Use of Standard Practices and Tools

The SEP to be used for Florida ITS projects is based on the subprocesses and tasks presented in the *IEEE 1220-1998* standard, tempered by actual experience in developing hardware/software systems for many different industries, including ITS. The *IEEE 1220-1998* standard presents the detailed requirements of the SEP, but indicates that an enterprise should tailor the activities of each task by adding or deleting activities, or tailor the subprocess tasks by adding or deleting tasks. *Section 6* of the *IEEE 1220-1998* standard details the SEP subprocesses that apply throughout a project's life cycle to all activities associated with project or system development, design, construction, verification/testing, training, operation, support, distribution, disposal, and human systems engineering. In general, the SEP is divided into the subprocesses identified below:

- Requirements Analysis
- Requirements Verification
- Synthesis
- Design Verification

Phase I of the SEMP project recommended the use of the ITS architecture and associated tools as part of a systems engineering approach to ITS projects. The following sections discuss the use of these standard tools in the SEP subprocesses.

It should be recognized that there are several COTS systems engineering tools that can support the various SEP subprocesses and tasks outlined in the *IEEE 1220-1998* standard. These tools were developed by the systems engineering community for users from different disciplines that want to implement the SEP. The Web site for the International Council on Systems Engineering (INCOSE) includes a database of these tools.[41] The INCOSE Web site allows several ways to view this tool database. One way is with the *IEEE 1220-1998* standard's SEP subprocesses. The user can select a specific SEP subprocess from a list presented on the site to view systems engineering tools that assist in the implementation of that subprocess. The user can examine these tools to determine if any meet their needs. These tools will not be discussed further in this document.

[41] The INCOSE Web site is available online at http://www.incose.org/.

CHAPTER 4 –
MANAGEMENT WITH THE SYSTEMS ENGINEERING PROCESS

4. Life-Cycle Management Using the Systems Engineering Process

The SEP is first a planning exercise, followed by the execution of the plan, accompanied by constant monitoring of the process. The program/project plan establishes a road map for the project team to follow and is the way the ITS deployment is fielded. Gen. Dwight D. Eisenhower was quoted as saying, "In preparing for battle, I have always found that plans are useless, but planning is indispensable." That is true of any planning exercise, but it is critical to have thought through the entire process of the project, and identified the standards and documentation needed to guide the engineers through the plan's execution.

The classical SEP is defined in detail in *Chapter 3* of this document.

For purposes of the *Florida's Statewide SEMP*, a "program" is defined as a long-term endeavor composed of one or more projects. It may be multiphased and funded incrementally. A "project" is the effort to accomplish a single ITS deployment. This section of the *SEMP* will focus on the planning aspects of a project.

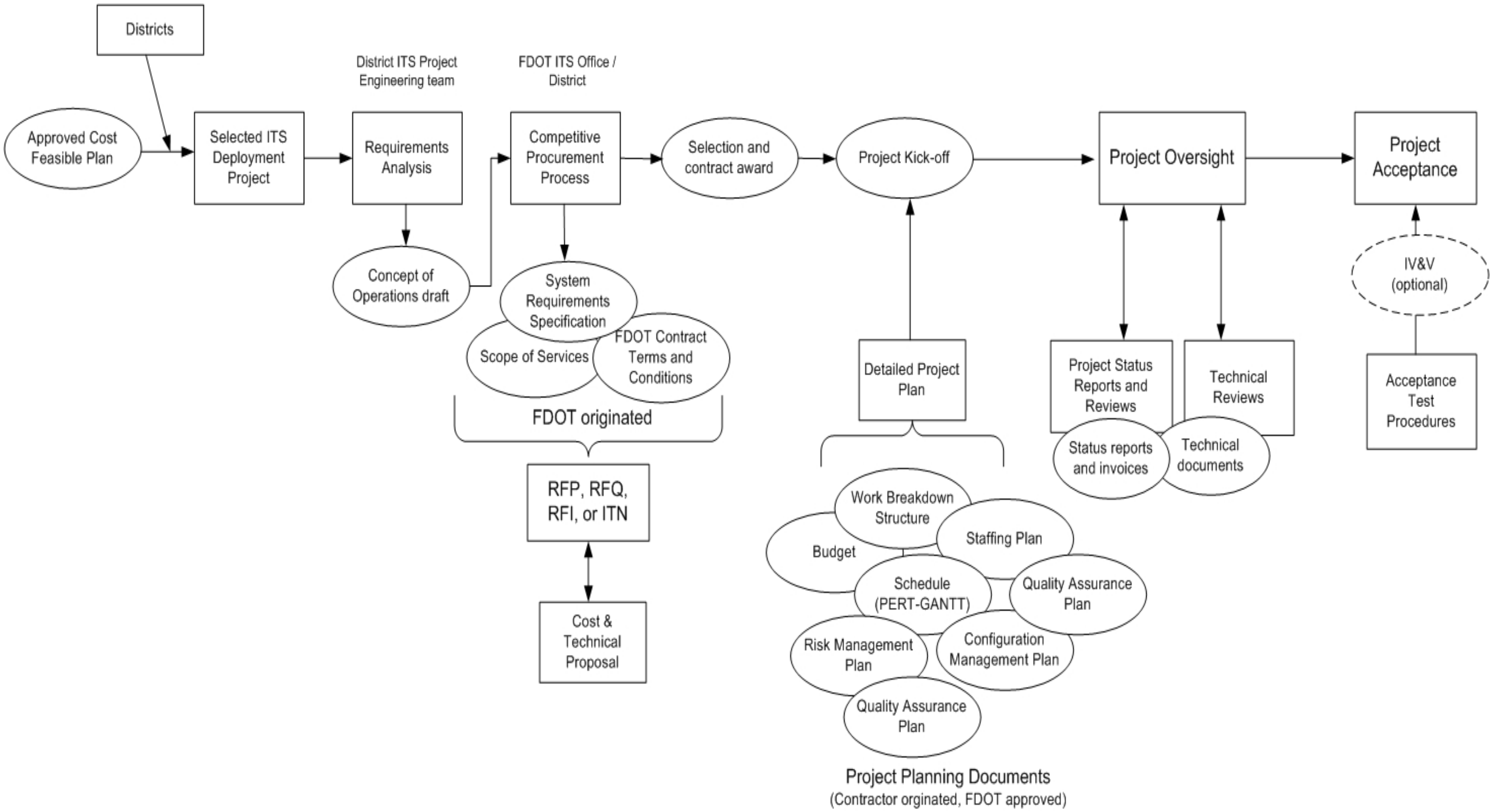
4.1 Scope

Florida's Statewide SEMP defines the interdisciplinary tasks required throughout a system's life cycle to transform customer needs, requirements, and constraints into an ITS solution. Before engineering can begin, the project plan must be developed and approved. This section of the *SEMP* will describe the steps necessary to plan a project, provide templates for the project planning documentation that is required to support the project, and offer guidance on managing change after the project starts.

4.1.1 ITS Project Model

The planning process described in this chapter is based on a typical FDOT ITS deployment where the FDOT contracts for goods and services that meet specified requirements. An overview of the project model is shown in Figure 4.1.

Figure 4.1 – ITS Project Model



The process starts when an ITS project is selected for deployment. The FDOT will choose an administrator and identify the particular project to be developed. The ITS project team is made up of FDOT managers and engineers, and may be augmented by outside consultants. The team identifies the stakeholder needs, and extracts or develops the system requirements. This is the start of the project planning process. The ITS project engineer identifies the overall objectives for the plan, then leads the effort to capture the project's vision through a draft ConOps.[42] (Refer to *Chapter 5* for a detailed template to use when writing a ConOps.)

The ITS project team describes the overall project plan, identifies contract deliverables, and spells out other project management requirements in a scope of services document. The team also develops a technical requirements specification for the ITS deployment. (Refer to *Chapter 4*.) The FDOT CSO is consulted to select the procurement method and contract type. The description of contract requirements is contained in the contract terms and conditions documentation. This documentation is needed to support the procurement process and are the basis for the contractor's project plans.

The contractor's proposal is expected to contain a description of how the contractor will manage the project according to FDOT requirements. After contract award, the contractor is expected to submit drafts of several project-planning documents shown in Figure 2.1. Prior to project kick-off, the FDOT and the contractor should finalize their interpretation of the system requirements, project plan, and reporting requirements. Contract documentation must be revised to reflect this understanding; however, care must be taken not to change the scope of services or system requirements radically, least the basis for the contract award be challenged. The contractor explains their project approach at the kick-off meeting, and it is here that the FDOT delivers the project development responsibility to the contractor, and begins project monitoring and oversight.

The acceptance test marks the transition in the system's ownership from the contractor to the FDOT. At this point, the contractor typically assumes a support role while the FDOT enters the operational phase of the project.

4.2 Purpose

The purpose of this chapter is to provide a guide to the ITS project engineer and other management proposal when developing a standardized project plan for ITS deployments in Florida. This document:

- Defines the critical elements of a project plan

[42] A concept of operations is referenced as the ConOps, CONOPS, and COO in different standards.

- Defines a set of standard practices and technical tools for use in planning, monitoring, and managing a project

- Establishes a formal process for planning and managing change on projects

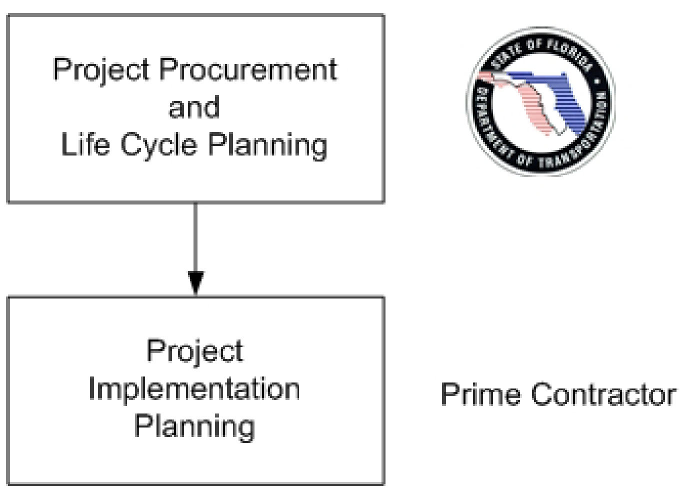
4.2.1 Planning Templates

The appendices to this document contain templates an ITS project engineering team can use to produce standard planning and project control documentation. *Chapter 8* provides guidelines on selecting which documents to include in the plan to fit the needs of a particular project.

4.3 Project Management Planning

When FDOT administrators select an ITS project for deployment, they start planning for it. The system's life cycle is considered. The environment it will operate in and the resources needed to support it are identified. There are two levels of planning that take place on typical ITS deployment projects, as shown in Figure 4.2. One is the planning the FDOT does, both at the District level and at the ITS Section level; the other is that done by the prime contractor if a system deployment is procured.

Figure 4.2 – Planning Responsibilities



4.3.1 Plan and Organize

A project's scope and risk determine the level of documentation needed and the amount of insight the FDOT needs into the development process. Typically, software projects carry a higher risk than do hardware procurement projects. Systems that require integration of hardware and software from more than one contractor are probably the highest risk projects and, therefore, need very careful planning and monitoring.

The basic premise for planning a project is to establish a project baseline in terms of schedule and deliverables; identify the single critical path for the project; and then manage and verify progress. Milestone demonstrations are a good way to do that.

4.3.1.1 Milestone Demonstrations

For integrated systems where functional capabilities are clearly identified, demonstrations of key functions at the deployment site are an excellent way to reduce risk, showcase early winners, and verify that progress is being made. Planning for milestone demonstrations requires a good understanding of the system's functional architecture as defined by stakeholder needs. By writing a draft ConOps before the procurement, the FDOT will identify the system functional capabilities that are needed to satisfy the stakeholder requirements. A milestone demonstration takes place at an opportune time in the contractor's integration of the system. This occurs when the contractor has assembled the pieces of the system and it is operating in the development facility. It requires that the contractor use a phased integration approach that follows functional threads of the system to benefit both the FDOT and the contractor. The FDOT gets to see parts of the system in actual operation long before formal acceptance, and the contractor gets to identify installation and system problems early while there is time and money left to fix them. A phased integration approach results in a more reliable system because it builds on previous versions of the software, so by the time the system is accepted, it has been tested a number of times and most of the bugs have been identified and fixed.

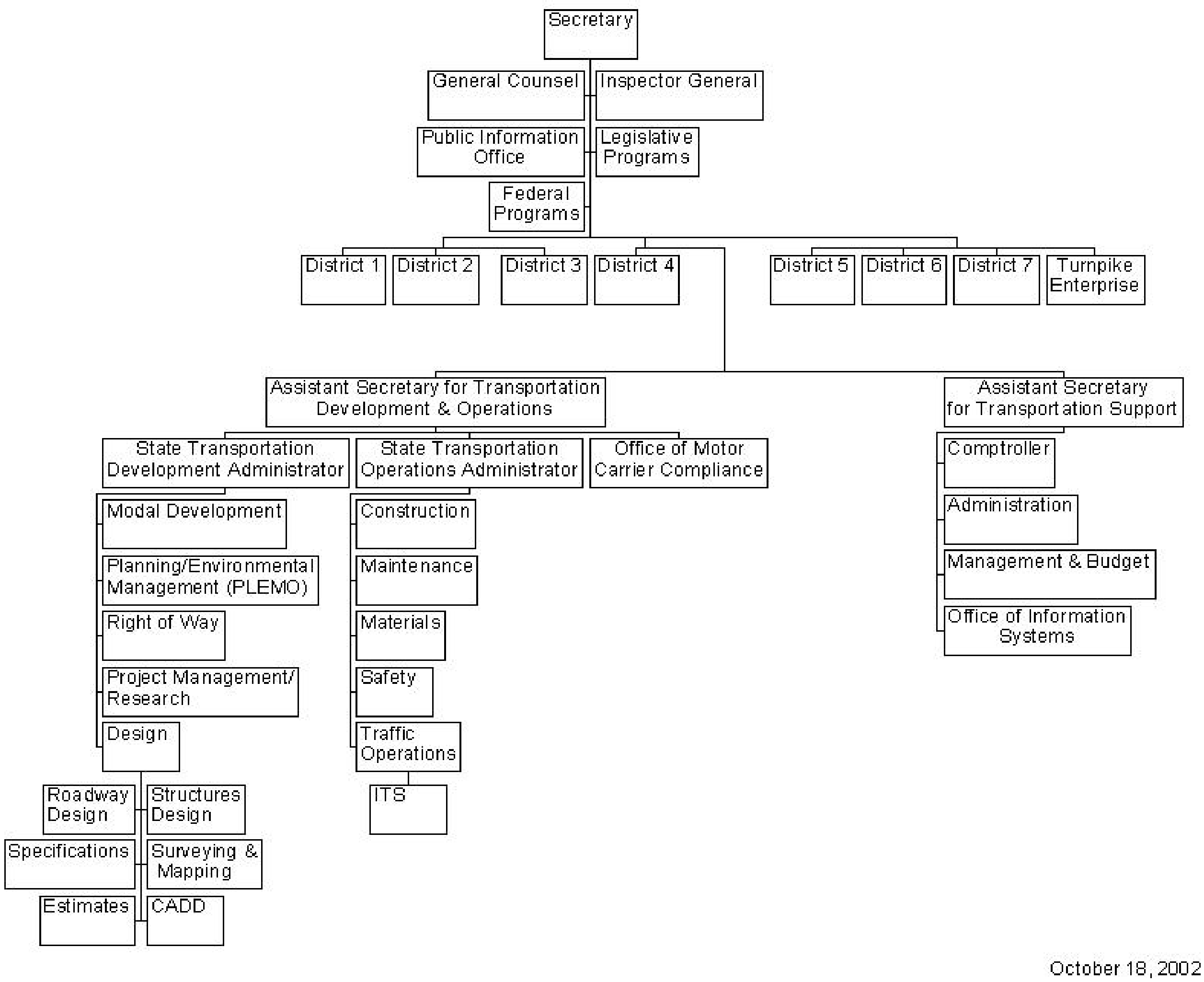
A milestone demonstration is supposed to be an informal demonstration of the system's capabilities and should not impose much additional effort on the contractor, since it should be part of their I&T process. Contractors may be uncomfortable doing "public" demonstrations of their partially completed systems, but this can be overcome if a good trust relationship has been established between the FDOT and the contractor, and if expectations are carefully managed.

It should be noted that milestone demonstrations apply pressure to both the FDOT and the contractor to have everything needed to support a scheduled demonstration ready. If a scheduled milestone demonstration slips, the reasons will be obvious and an accurate assessment of the project's progress will be possible. In some high-risk, high-profile projects, it may make sense to tie milestone demonstrations to progress payments and award incentives for early, successful demonstrations of key system capabilities.

4.3.1.2 FDOT Organization

The organization of the FDOT as of October 2002 is shown in Figure 4.3.

Figure 4.3 – FDOT Organization Chart



4.3.1.3 Negotiating Technical Commitment

The FDOT shall designate an individual, typically the ITS project engineer, with authority for negotiating technical commitments on behalf of the FDOT Traffic Engineering and Operations Office (TEOO) ITS Section. The designated FDOT individual can delegate the authority to others to act on the FDOT's behalf, as applicable. In the TEOO ITS Section, this individual is called an ITS administrator; at the District level, that individual may be called the project engineer.

4.3.1.4 Critical Resource List

A critical resource list (CRL) will be prepared and maintained upon the issuance of a contract or task work order (TWO) for an ITS project's implementation. The CRL reflects the agreement between FDOT stakeholders on providing and receiving resources. Resources can be any of the following:

- Documentation
- Hardware (including throughput)
- Software
- Facilities
- Staffing
- Joint operations procedures
- Data

A resource is defined as critical if it is required for a critical path activity. The CRL includes a minimum of the following information:

- Item(s) to be provided
- Source(s)
- Recipient(s)
- Date needed
- Acceptance criteria, as applicable

The need dates and availability dates shall be tied to the project schedule.

The FDOT prepares the CRL, so it is not a contract deliverable. The contractor, however, is required to provide a staffing plan, and may furnish information on resources and capabilities in the proposal. The CRL reflects the necessary institutional agreements and infrastructure needed to support the implementation and operation of a project.

The ITS project engineer is responsible for generating and maintaining the CRL. The FDOT ITS Section's project administrator is responsible for gaining agreement between sources and receiving groups, and for providing the necessary liaison to share resources and coordinate schedules. If a project is implemented solely by a District with no need for outside resources, then the ITS project engineer is responsible for coordinating the CRL.

Contingency planning for resources that may be unavailable is incorporated into the risk management process. The CRL shall be located and maintained in the ITS Section where the ITS project administrator resides. On large projects, the ITS project administrator may be assisted by a project planner, who is responsible for querying the ITS project engineer and appropriate task leaders periodically for status and changes, including comparisons of the status and actual or projected completion dates to the plan, and the impact of late or early completions. It is strongly recommended that status be monitored weekly, biweekly, or daily if critical path activity, such as final on-site system I&T, is taking place.

4.3.1.5 Contract Work Breakdown Structure

The contract work breakdown structure (CWBS) is often referred to as the work breakdown structure (WBS), and its purpose is to show the way work tasks are organized. Budget detail should relate directly to the CWBS and the work should be monitored according to the CWBS.

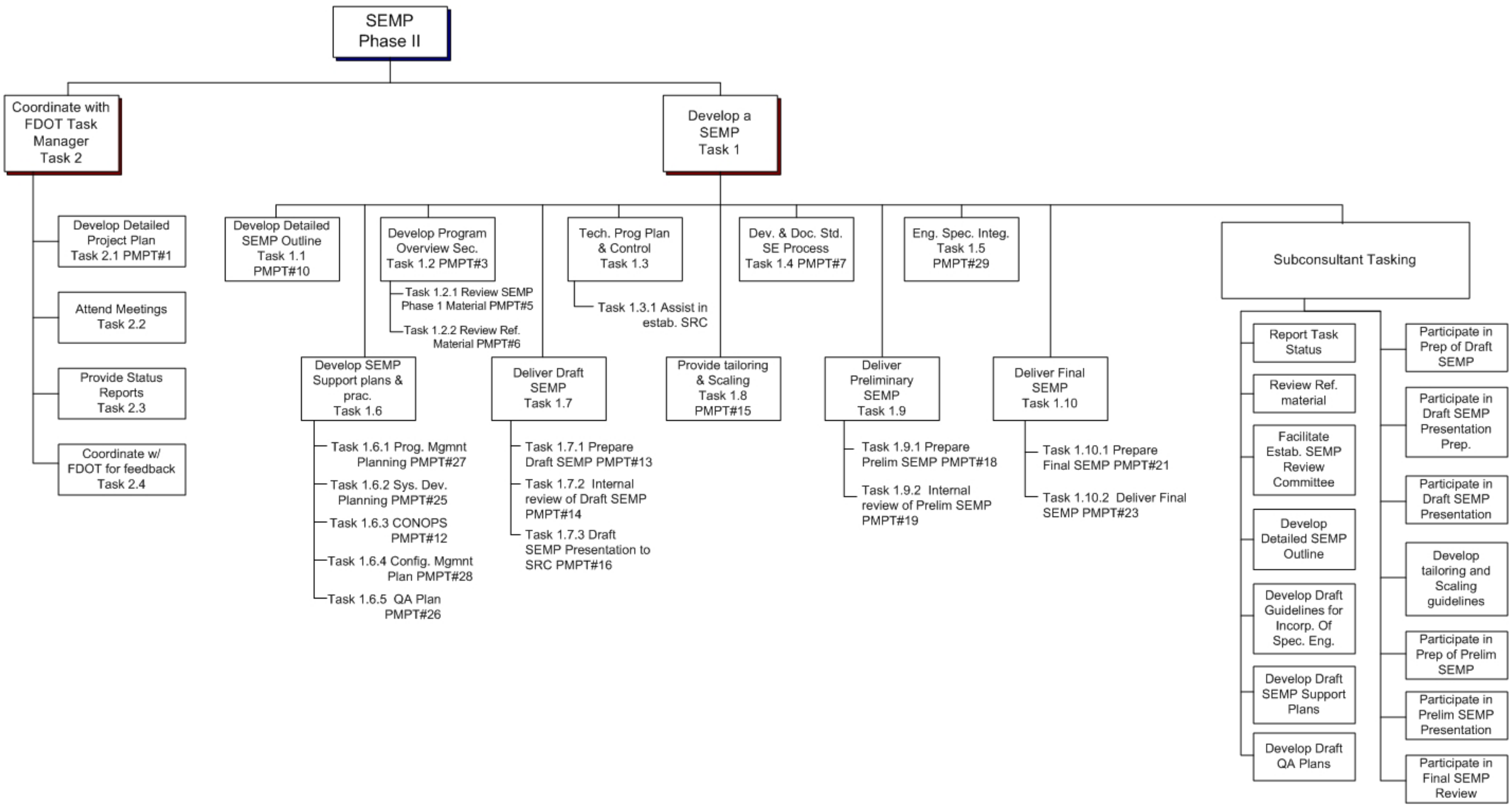
The preferred way to show the CWBS is in graphical form followed by a paragraph describing the work involved in each task shown. A task numbering scheme should follow the hierarchy shown in the graphical structure. Figure 4.4 is a graphic depiction of a CWBS for the project that developed this document.

Changes to the CWBS should not be made unless a contract change order is issued. The CWBS is first drafted in the scope of services document. The contractor provides a more detailed version, or an interpreted version is contained in the detailed project plan.

4.3.1.6 Project Life Cycle

A detailed explanation of the project life cycle is provided in *Section 7.1*.

Figure 4.4 – Graphical Depiction of a Work Breakdown Structure



4.3.1.7 Work Products

This section discusses the required work products that result from the application of the FDOT *SEMP*.

All work products (i.e., documentation, data, code, etc.) that one engineering group produces for another engineering group's use are reviewed and approved by the receiving group. For the FDOT, the contractor prepares the documentation, and the FDOT reviews and approves it. This submittal and approval process is documented using a contract deliverable requirements list (CDRL) that specifies the requirements for each deliverable, when it is due, and what the contents should be. In some cases, the FDOT creates a data item description (DID) for each deliverable that prescribes the format and content for each work product. The CDRL and the DID are part of the procurement process.

4.3.1.7.1 Systems Engineering Project Planning Documentation

All technical plans undergo a coordination review. The review ensures consistency with high- and low-level plans according to the document tree included herein. All project plans include provisions for changing the plan and/or deviating from the plan. Each plan includes a revision history panel that is used for recording information about when the document was created and for tracking subsequent changes. An example of a recommended revision history panel is shown in Figure 4.5. This panel can be copied and pasted into the Microsoft Word document being created.

The following is a sample list of project work products; the tailoring guidelines will recommend which ones to include based on the project scope. Templates are provided in the appendices for use in developing a standard set of project planning documentation. Sample work products include:

- **Systems Engineering Management Plan (SEMP) –** This document is a tailored version of *Florida's Statewide SEMP*, and is applicable to the specific project. (Refer to *Chapter 8*.) The FDOT ITS project manager writes this plan and it must be produced before the procurement process starts.

- **Scope of Services –** Sometimes called the statement of work (SOW), this document lays out the general work required to deploy the specific ITS project. The FDOT ITS project manager writes the scope of services.

- **Detailed Project Plan –** Prepared in response to the contractual scope of services document, this document is created by the contractor after a contract is issued. This document can contain the detailed project schedule and staffing plan, or they can be separate submittals.

Figure 4.5 – Sample Recommended Revision History Panel

Revision History Panel			
File Name:			
File Location:			
Document Tracking No.:			
CDRL No.:			
Version No.:			
Revision No.:			
Name		**Initial**	**Date**
Created By:			
Reviewed By:			
Modified By:			
Completed By:			

- **Detailed Project Schedule –** This document is created using Primavera and must be based on a closed network of tasks. See the description of the project evaluation and review technique (PERT) provided in *Section 5.1.2.1* to see how it is created and used. Most scheduling software that supports PERT can show the schedule in the GANTT format. The value in using PERT is that the critical path is easily identified and the ITS project manager can pay particular attention to tasks along the critical path to make sure the project doesn't slip its completion date.

- **Software Development Plan (SDP) –** Created by the contractor developing the FDOT's software, this document explains how the software will be managed, its structure, and what metrics will be used to measure progress. The SDP identifies who is responsible for developing software, who is responsible for integrating the software modules, and who is responsible for testing the software. The CM procedures are discussed, along with software quality assurance (SQA), even though there are separate documents published to govern the SQA process and the software configuration management (SCM) process.

- **Hardware Development Plan (HDP) –** The HDP is used less often than the SDP because hardware engineering generally follows a standard process and it is easier to verify that progress is being made. However, should the ITS project manager need to monitor the development of custom hardware, the contractor should develop a plan on how the hardware will be developed, integrated, and tested to verify that it meets the FDOT's requirements.

- **Configuration Management Plan –** This document can be produced by both the FDOT ITS project manager and the contractor. It is project-specific and should show how the project's CM process integrates with the state's CMB. The contractor should also create a CM plan for the project that discusses how they will adhere to the FDOT's CM requirements and how they will manage the system's configuration up to the point where the FDOT accepts it.

- **Quality Management Plan (QMP) –** This document can be produced by both the FDOT ITS project manager and the contractor. It is project-specific and should show how the project QM process conforms to the *Statewide Quality Assurance Plan for ITS Deployments*.[43] The contractor should also develop a QMP to show how they plan to manage quality internal to the project. Refer to *Section 5.2* for more information on QM planning.

[43] Bonds, John M. (PBS&J), *Technical Memorandum 3.1: Statewide Quality Assurance Plan for Intelligent Transportation System Deployments, Draft Version 2.1*, FDOT Contract No. C-7772 (February 2004).

- **Training Plan –** The contractor develops the training plan in conjunction with the FDOT ITS project manager. It explains how training will be accomplished, what the training objectives are, and the logistics involved with the training. The training schedule is coordinated with the project's master schedule and considers the availability of FDOT resources.

- **Software Security Plan –** Typically, software is seen as a way for outsiders to gain access to and control of an FDOT ITS service. Every software development that has such a potential must have a security plan. The security plan describes the hardware, software, and procedures that need to be implemented to provide a secure system.

- **Facility Security Plan –** Contractors designing and building high-value FDOT facilities need to consider physical security and how to control access to the facility. This plan identifies probable threats to the facility, and specifies what procedures and security systems are to be used to mitigate the perceived threat. A cost-to-benefit analysis of what it will cost to mitigate a given threat versus the probability of that threat occurring is also contained in a facility security plan so that the budget for project security can be allocated effectively.

4.3.1.7.2 Systems Engineering Work Products

Planning documentation describes the plan for implementing a project. Engineering documentation is the result of executing the project plans. The following is a sample list of project work products; the tailoring guidelines will recommend which ones to include based on the project scope. Templates are provided in the appendices for use in developing a standard set of project engineering documentation.

- **Requirements Database –** All project requirements should be managed using a database that provides change control information as well as relating information about the source of the requirement, such as the system specification paragraph, stakeholder interview date/time, etc. The database should also be capable of showing the hierarchal structure of the system requirements so that any requirement can be traced back to a user need. More information on requirements databases is provided in *Section 4.2.1.2*.

- **System Design Document (SDD) –** The SDD is a description of the system from a functional viewpoint. It identifies the parts of the system in a hierarchal fashion using the black-box technique of system functional decomposition. (Refer to *Section 4.2.1.1*.) The SDD will describe the external interfaces to the system, and the major internal interfaces between subsystems and components. The contractor develops the SDD, which shows how the contractor's design solution meets the system functional requirements. The SDD is required after the system specification is approved and before the FDR or CDR.

- **Software System Design Document (SSDD) –** The SSDD is a variant of the SDD. The SSDD describes the software system's design and shows how the software design meets the system requirements. On projects whose only product is software, the SSDD is often simply called the SDD.

- **Interface Control Specification (ICS) –** External interfaces to the system are specified in this document. It lists the requirements for each external interface that include data format and protocol. The contractor designing and integrating the system typically writes the ICS.

- **Interface Control Document (ICD) –** When the system is accepted and the interfaces have been proven to work as specified in the ICS, the ICS becomes the ICD and is used to control the interfaces. No changes to the ICD may be made without approval of the CMB.

- **Concept of Operations –** It is recommended that the FDOT ITS project engineering team develop a draft ConOps for each major project in response to stakeholder needs. The ConOps describes how a hypothetical system will work from the viewpoint of operators, managers, and users of the system. The contractor is expected to take the draft and revise it by adding detail to show how their design will meet the needs of the stakeholders. *Chapter 5* contains a detailed explanation of the ConOps' role and how it is created.

- **Test Plan –** Although a planning document, the test plan is a product of the engineering effort that describes how the system under development will be integrated and tested. The test plan is created by the system's developer; identifies ICs and TCs; and provides the RTVM. Refer to *Section 4.3.2* for information on the RTVM and to *Section 4.3.3.1* for more detail on writing a test plan.

- **Test Procedures –** This document describes the systematic procedures to follow to accomplish the test plan and should contain test data sheets that are used to conduct the testing. The document is created by the group who will integrate and test the system. Typically, development of the test procedures is an iterative process using trial and error to arrive at a set of procedures used to verify that the system meets the requirements. During the course of developing the test procedures, system bugs are discovered as well as misconceptions about how the system should operate. The ITS project manager should plan on three cycles of trial and error for each test procedure before a satisfactory set of procedures is arrived at. *Section 4.3.3.2* contains more information on how to write a test procedure.

- **Test Reports** – When a formal test procedure is completed, the test data sheets are collected, and a formal report is written to document what happened during the test and to recommend actions if any requirements were not completely satisfied. Test reports should also contain a brief analysis of what caused the system to fail to meet a requirement. Refer to *Section 4.3.3.3* for information on how to write a test report.

- **Trade Studies** – Sometimes called cost-to-benefit analyses, trade studies are used to determine the optimal course of action when developing a system. They are called trade studies because a trade-off analysis is done between viable alternatives to a design solution. Cost is often a primary factor in determining the best solution, so a comparison of each solution's cost weighed against its benefit is needed. It is good engineering practice to consider alternative designs when creating the system architecture. Refer to *Section 4.2.1.4* for more information on how to do trade studies and when they are recommended.

- **White Papers** – A white paper is a study of a particular issue developed by either the FDOT or the contractor to gather the facts in one document, analyze the issue, and provide a recommendation or alternate choices of action. A white paper is usually developed in support of a decision process. Typical white papers average 10 to 20 pages in length.

- **Point Papers** – Like a white paper, a point paper is a summary of the key issues and facts, along with the author's recommendations. It is much shorter than a white paper and typically averages one to two pages.

4.3.1.8 Peer Reviews

The peer review process aids in identifying and removing defects from engineering products as early in the technical effort as possible. All of the systems engineering work products identified in *Section 2.1.7.2* above should be checked for technical accuracy by at least one technical peer, and for spelling, syntax, and grammar by an editor.

4.3.1.9 Systems Engineering Tools

This section provides a list of the primary tools used in the SEP and briefly describes the main functions for which each tool is used. By using this set of primary tools, the FDOT will have the ability to share work products across all of the FDOT. The SEP's primary tools include:

- **Microsoft Word** is the primary document creation tool. Standard templates are provided in *Florida's Statewide SEMP* for most of the engineering work products identified. The TEOO ITS Section provides guidelines on how to format and apply Microsoft Word. Refer to the *FDOT ITS Office Written Business Communications Style Guide*,[44] which is available through the TEOO ITS Section in Tallahassee.

- **Microsoft Excel** is the primary spreadsheet creation tool used to analyze costs or performance numbers. Excel spreadsheets can be imported as tables into Microsoft Word or tables from Word can be imported into an Excel spreadsheet. It is recommended that all data be included within a single Excel workbook. Typically, when an Excel workbook is created, it will contain three data sheets. If only one sheet is needed, it is recommended that the two blank sheets be deleted. If macros are used in the workbook, provide sufficient comments in the spreadsheet to explain how the macros are used and how they work. Lock any cells that, if changed, would materially change the results of the spreadsheet.

- **Microsoft Visio**®[45] is the primary drawing tool used to illustrate engineering work products. Although a Visio drawing can be copied and pasted directly into a Microsoft Word document, it is recommended that a JPEG copy of the drawing be used so the drawing cannot be changed and is more portable.

- **Microsoft PowerPoint**®[50] is the primary tool used for presentations. A good rule of thumb is to keep things simple. Avoid putting too much detail or text on a single slide, and be sure to use a font size of 16 points or larger to ensure legibility. Limit your use of animation because it can be distracting and does not reproduce well in printed copy. Avoid using audio or video clips unless necessary. Choose simple backgrounds that are light in color so your slides will reproduce better on black-and-white copiers.

- **Primavera**®[46] is the FDOT's primary planning and scheduling tool used. It is recommended that the PERT network format be used to create the schedule and the GANTT format be used to show the schedule. Often, it is necessary to use Visio to create a simplified version of the GANTT schedule for presentations since the Microsoft Project tool typically provides a great amount of detail that spans a large space, making it hard to include in project documentation.

---

[44] PBS&J, *Florida Department of Transportation Intelligent Transportation Systems Office – Written Business Communications Style Guide*, FDOT Contract C-7772 (October 2003).

[45] Visio, PowerPoint, Microsoft Project are registered trademarks of Microsoft Corporation in the United States and/or other countries.

[46] Primavera is a registered trademark of Primavera Systems, Inc.

- **Microsoft Project®**[50] is an inexpensive planning and scheduling tool that may be used as an alternate to Primavera so long as is used with a supplemental software tool call **PERT Chart Expert™**[47] (PCE). PCE adds functionality to Microsoft Project that allows easier creation of PERT charts and allows them to be converted to files that can be imported into Primavera.

- The **FileMaker Pro®**[48] requirements database is a relational database that is has been customized for use on FDOT projects to manage requirements. It is used as a flat file database to manage system requirements. Refer to *Section 4.2.1.2* for an explanation of how the requirements database is created and used.

- **IBM®**[49] **Rational®**[55] **RequisitePro®**[50] is a tool provided by IBM's Rational Software Division and is an integrated product for requirements management. It promotes better communication, enhances teamwork, and reduces project risk. It is a more automated tool than the FileMaker Pro tool and is integrated with other IBM Rational software support tools;

- **IBM Rational ClearCase®**[55] is used to manage software configuration. A product of IBM's Rational Software Division, ClearCase provides software asset management with integrated defect and change tracking;

- The **IBM Rational ClearQuest®**[55] tool is used to manage software changes by providing activity-based change and defect tracking. It works with Rational ClearCase to provide a complete SCM solution.

- The FDOT has selected **Oracle®**[56] as the standard for ITS projects using databases. **Oracle9*i*0 Designer**[51] is a toolset used to model, generate, and capture database requirements. It models the design of these databases and their applications. It also allows the developer to assess the impact of changing those designs or applications. This tool will typically be used by the software developer; and

[47] PERT Chart Expert is a trademark of Critical Tools, Inc.

[48] FileMaker Pro is a registered trademark of FileMaker, Inc.

[49] IBM is a registered trademark of International Business Machines Corporation.

[50] Rational, RequisitePro, ClearCase, and ClearQuest are registered trademarks of Rational Software Corporation in the United States and in other countries.

[51] Oracle and Oracle9*i* are registered trademarks or trademarks of Oracle Corporation.

- The FDOT is participating in the use of a GDT® **Dynamap™**[52] statewide license for GIS map data that uses a proprietary technology called Dynamap. All GIS map products from the GDT product should be capable of exporting or importing shape-type files.

4.3.2 Integration of Disciplines

This section discusses the different engineering teams needed on a project and their role in the systems engineering process.

4.3.2.1 Systems Engineering Integration Team

The objectives of the Systems Engineering Integration Team (SEIT) are to:

- Integrate across engineering disciplines
- Integrate across ITS products
- Establish and maintain system requirements
- Establish and control system baselines
- Specify, document, and control external and internal interfaces
- Conduct system integration and acceptance testing
- Manage the production and control of technical documentation
- Manage technical risks
- Charter working groups to work specific program-wide technical issues

The full SEIT organization for typical projects is shown in Figure 4.6. If some disciplines are not used on the project, the function can be deleted. It assumes that an ITS deployment is managed by the District. The TEOO ITS Section provides coordination, standardization guidance, and, on occasion, funding. If the ITS Section has a GEC, they may provide advisory support through the ITS Section.

The District engineer should appoint a project engineer who will be responsible for the technical and administrative aspects of the project. On large projects, the ITS project engineer may appoint a chief project engineer to manage all technical aspects of the project. If a contractor is hired to implement the project, the engineering team shown beneath the chief project engineer will be the contractor's team. The District may assign counterparts from the engineering staff at the District, if needed.

[52] GDT and Dynamap are registered trademarks of Geographic Data Technology, Inc.

Each District is expected to implement an independent quality assurance (QA)/quality control (QC) function that reports directly to the District engineer and monitors the engineering effort on each project to assure that quality is being planned and managed. The District ITS project engineer is also assisted by District support functions, such as legal counsel, procurement, contracts management, and other administrative support functions. If these functions are not available at the District level, they may be provided through the ITS Section in Tallahassee.

The basic project team that is developing a hardware and software ITS deployment is composed of systems engineering, hardware engineering, and software engineering. The chief project engineer is selected from one of the three engineering specialties, based on the nature of the project (i.e., is it predominately a hardware procurement or a software development?), and the experience and seniority of the individual. The chief project engineer must have strong leadership skills, a working knowledge in all engineering disciplines used on the project, management experience, and excellent communication skills. On small projects, the senior systems engineer may fulfill the role of chief project engineer or the ITS project engineer may double as the chief project engineer.

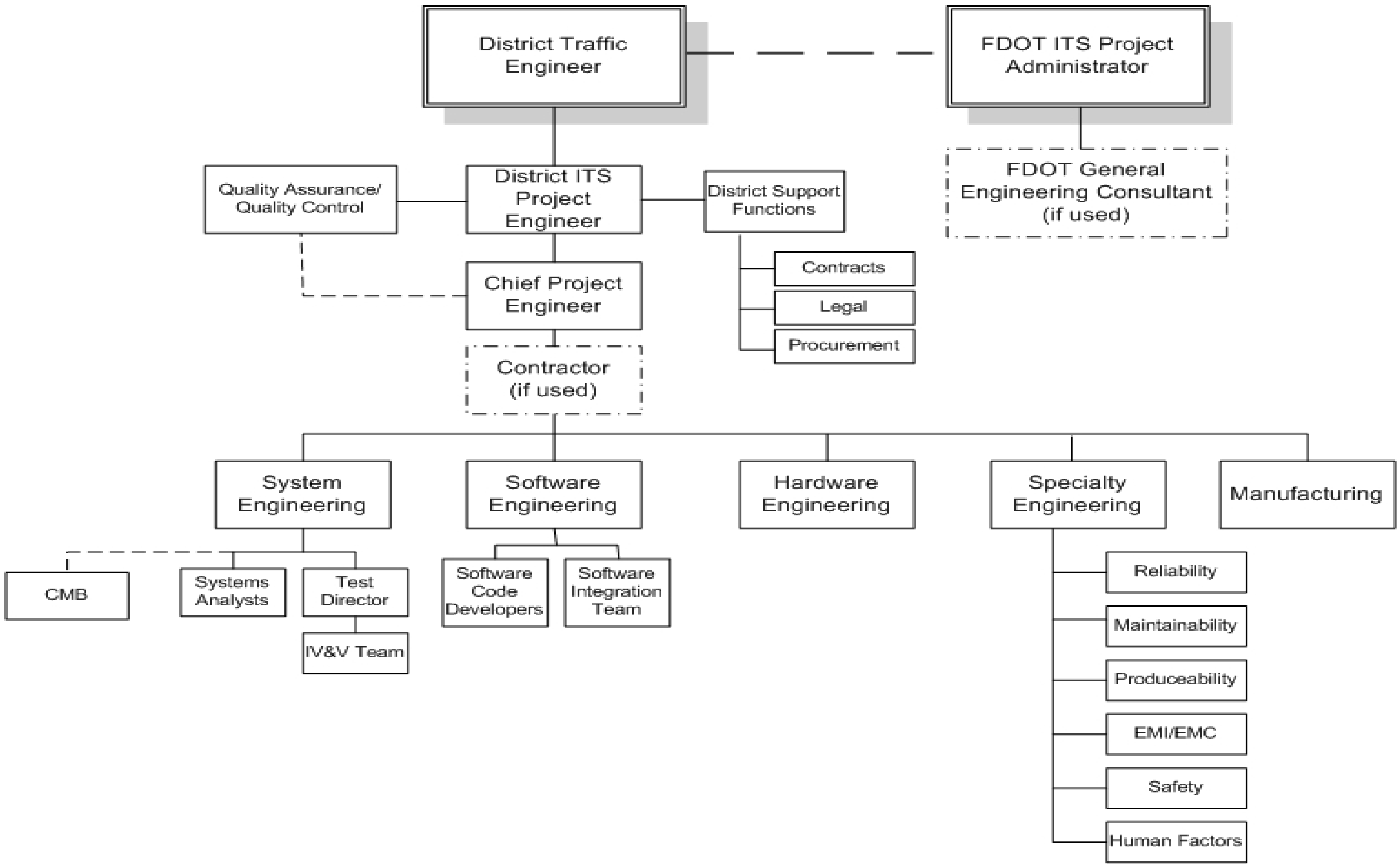
The systems engineering group is usually composed of the system analysts, who focus on the front-end engineering effort, and the IV&V test team. It is good practice to assign an experienced engineer the job of test director, responsible for all aspects of I&T. The IV&V team must be truly independent of the team developing the ITS product, and may be provided through the ITS Section or by an independent contractor.

The software engineering group consists of software developers and a team of experienced software engineers responsible for integrating the software modules, testing them, and maintaining the software configuration.

The hardware engineering group consists of engineers familiar with the system's hardware components and the hardware interface. If the project is designing and manufacturing hardware units, the hardware engineering group would consist of many engineers, but on projects that are predominately software development, the hardware engineering group may consist of one engineer.

Both hardware and software engineering groups seek guidance from the systems engineering group, which maintains the overall system requirements. Each group is expected to derive its requirements from the system requirements. The systems engineering group is ultimately responsible for integrating the hardware and software into a system that meets the requirements.

Figure 4.6 – Organization of a Typical SEIT



The specialty engineering group provides the unique expertise for how to design a system that is reliable, easy to maintain, and easy to fix when it breaks. It also provides design expertise to make sure the system under development is designed so as not to interfere electronically with other systems, referred to as electromagnetic interference (EMI), and to be electromagnetically compatible (EMC) with other systems in the proximity of the one being deployed. If there is a large amount of user interface, then the engineers who are expert in human factors engineering (HFE) design the system to meet the needs of the human operator. If the system could be hazardous to life and property, such as a microwave relay station with high-powered radiation transmissions or large pieces of moving machinery, then the safety engineer makes sure that the system is designed to minimize opportunities for personnel to be injured or property to be damaged.

Manufacturing engineering is staffed if there will be a large number of units manufactured. The focus in this group is on designing the system so that it is economically reproducible. Examples of solutions provided by this team are a reduction in the number of parts in a system; the placement of parts for ease of assembly; and color coding connectors or designing unique connector shapes to prevent wrong assembly of parts.

4.3.2.2 Issue Resolution

This section describes the procedure and mechanism by which project issues are documented, tracked, and resolved. All projects should establish a problem tracking report (PTR) system and an action item tracking database. All technical issues that are raised during the development of the system are entered in the PTR system. Problems are assigned to project team members for resolution through action items and the ITS project engineer is responsible for ensuring that problems are addressed in a timely manner. The minimum information in the action item tracking database includes the:

- Action item identification numbers. It is recommended that a sequential number based on the month and year when the action is assigned be used, for example, MMYY-001.
- Description of the problem
- Who issued the action item
- Who the resolution of the action item is assigned to
- The expected date the action should be resolved
- The current status of the action item (i.e., open, closed, or no longer applicable).

The PTRs are used predominately by the SEIT during I&T. They are similar to the action item tracking database, except they contain more technical details and analysis.

4.3.2.3 Intergroup Critical Dependencies

All intergroup critical dependencies are clearly identified on the integrated master schedule and show the critical dependencies that include:

- Critical items, such as data, hardware, software, etc.
- Supplying group
- Receiving group
- Dates
- Acceptance criteria, if other than normal unit testing/receiving inspections are required
- Status of the system level schedule provided for project reviews

4.4 Procurement Process

Florida statutes and administrative rules prescribe a formal process to follow for acquiring products or services in accordance with the planned technical effort and requirements.

4.4.1 Acquiring Professional Services

This information is based on professional services (i.e., engineering, architecture, landscape architecture, and surveying and mapping) acquired under *Section (§) 287.055* of the *Florida Statutes*[53]; *Rule 14-75* of the *Florida Administrative Code* (*FAC*)[54]; and the FDOT's *Procedure 375-030-002*.[55] There are seven steps in the process:

- Letters of response are received.

- The ITS project manager long lists to 10 consultants.

- The selection team or committee shortlists to three consultants.

- The technical proposals are reviewed.

- The selection team ranks firms 1, 2, and 3 based on published selection criteria.

[53] § 287.055, F.S., *Consultants' Competitive Negotiation Act*.

[54] FLA. ADMIN. CODE R. 14-75, *Qualification, Selection and Performance Evaluation Requirements for Professional Consultants to Perform Work for DOT* (April 2003).

[55] Florida Department of Transportation, *Procedure 375-030-002 – Acquisition of Professional Services* (February 2004). More information regarding the FDOT's Professional Services is available online at http://www.dot.state.fl.us/procurement/doingbus.htm#commodities.

- A contract with the first ranked firm is negotiated. If no agreement is reached, negotiations with the second ranked firm begin, or the procurement is cancelled and reissued.

- A written agreement or contract is developed.

4.4.2 Procuring Commodities or Contractual Services

The following procurement process is based on regulations that govern contractual services (i.e., all other services and commodities) as acquired under *§ 287.057* of the *Florida Statutes*[56]; *Rule 60A* of the *FAC*[57]; and FDOT *Procedure 375-040-020*.[58]

4.4.2.1 Invitation to Bid

If a procurement is estimated to cost less than $25,000, then the lowest responsive bid wins and a purchase order is created using Florida's electronic procurement system.

4.4.2.2 Requests for Proposals

If a procurement is estimated to cost $25,000 or more, the following process is followed:

- The project technical requirements, scope of work, and sample contract terms and conditions are established.

- The selection criteria is established and included in a request for proposals (RFP).

- The RFP is published.

- Separate technical and price proposals are received from all interested vendors.

- The selection committee reviews and grades the technical proposals,

- All vendors whose proposals meet or exceed 70 percent of the selection criteria have their price proposals opened. Price points are awarded according to the published selection criteria, with more points being awarded for the lower price.

- The most responsive vendor with the highest total technical and price points is selected.

---

[56] § 287.057, F.S., *Procurement of Commodities or Contractual Services*.

[57] FLA. ADMIN. CODE R. 60A, *Division of Purchasing*.

[58] Florida Department of Transportation, *Procedure 375-040-020 – Procurement of Commodities & Contractual Services* (May 2004).

- A purchase order is created using the electronic procurement system.

4.4.2.3 Invitation to Negotiate

Another procurement mechanism popularly used is the invitation to negotiate (ITN). The ITN is applicable for procurements expected to cost $25,000 or more. The following process is used for an ITN procurement:

- The project technical requirements, scope of work, and sample contract terms and conditions are established.
- The selection criteria is established and included in the ITN.
- The ITN is published.
- Letters of qualification from interested vendors are received.
- The letters of qualification are reviewed and three of the most qualified are selected for further negotiations.
- The technical and price negotiations are conducted with all vendors.
- The most responsive vendor that proposes the best value for the FDOT is selected.
- A purchase order is created using the electronic procurement system.

Samples of all the procurement methods are located on the Infonet site in the Procurement section. [59]

4.4.2.4 Best and Final Offer

If negotiations are between two or more firms, then their technical information must not be shared between firms by the FDOT negotiating team during negotiations. However, if it becomes apparent that neither firm's initial offer fully satisfies FDOT requirements, then the FDOT may request a best and final offer (BAFO) from the competing firms. The BAFO affords the competing teams with the opportunity to revise their technical and price proposals based on what they learned during negotiations.

[59] More information regarding the electronic procurement system is available online at http://infonet.dot.state.fl.us/OfficeWebSites.htm under the Procurement link.

It is to the FDOT's advantage to issue a BAFO because competing firms may view it as an incentive to improve their technical response or lower their price. Under no circumstances should a second BAFO be issued, since the procurement process would then take on the appearance of an auction to the lowest bidder. It should also be remembered that if a firm bids the project at a loss, they will be constantly looking for ways to increase the cost of the project and will be inflexible when the project's scope needs changes.

4.5 Risk Management

All projects must have a well-defined process for identifying risk and a methodology for managing it. The risk management plan should cover activities for all functional disciplines, as well as all CWBS elements. All project managers should review the project risk as an integral part of normal project reviews. Program risks must be identified and their mitigation actions planned.

Risk to a project can generally result in cost increases, schedule slips, resource limitations, or technical incompatibilities. Although it may be said that all risk results in cost increases, most projects identify risk using the following categories:

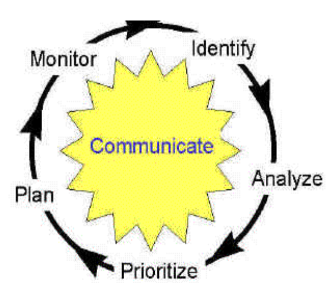
- Cost
- Schedule
- Technical
- Operational
- Organizational

4.5.1 Risk Cycle

The risk management cycle shown in Figure 4.7 runs continuously throughout the life of any project. A successful risk plan consists of the following components:

- Risk identification
- Risk analysis
- Risk prioritization
- Risk planning
- Risk monitoring

Figure 4.7 – Risk Management Cycle



4.5.2 Risk Identification

At the beginning of each project and periodically thereafter, the most experienced project team members should meet to brainstorm all the ways the project could fail. Each risk that is identified is placed in one of the five categories listed above. There is no ranking of the risk importance or likelihood at this time. It is important to identify specific risks and not "generic" risks. Generic risk is that which is generally true of most projects and a good project plan deals with those risks. It may be that initially there are no risks identified, but it is important to set up a process to identify and mitigate risk as the project evolves.

4.5.3 Risk Assessment

Each risk identified is evaluated for its potential to cause the project to fail. After identifying a broad range of possible risks, risks are quantified and prioritized in the assessment phase. At a minimum, each risk must be evaluated for its potential damage to the project in the category it is identified in and the likelihood of it occurring. It is recommended that the ranking of the damage it could cause and the likelihood of concurrence be simple and limited to three levels: high (3), medium (2), and low (1). Consequently, a risk will have two ratings: one for potential for harm and the other for the likelihood of occurrence.

4.5.3.1 Assignment of Priorities

The risk manager ranks each risk according to the likelihood of occurrence, and an overall risk ranking is then calculated by multiplying the likelihood of occurrence by the potential for damage to the project.

For each risk identified, the information listed below will be developed and recorded in a database or table that will serve as the basis for periodic risk management review. Risk information includes:

- A description and its probable outcome in terms of either the project cost or schedule
- Likelihood of the risk occurring (i.e., 1-low, 2, or 3-likely)
- Damage the risk can cause to the project (i.e., 1-low, 2, 3-great)
- Mitigation strategy
- Actions taken to mitigate the risk
- Current status of the risk mitigation efforts.

4.5.4 Risk Mitigation

This section describes the process of developing the risk mitigation plans. In this phase, existing risks are monitored and reported to the risk manager. Once there is no chance of the risk's occurrence, or if the time for the risk to impact the project has passed, the risk manager will change the risk status to "closed."

Typically, most risk identification, analysis, and planning takes place during a project's early stages. However, risk monitoring is a continual process. As new risks are generated and as additional information is learned, technical approaches modified, and personnel turnovers take place, new risks are identified or information is updated on existing risks. During each project status meeting, all open risk items will be discussed and the mitigation actions evaluated.

4.6 Assess and Evaluate Technical Effort

This section describes how to monitor and measure the technical effort.

4.6.1 Monitoring and Control

The ITS project team must institute the monitoring and control functions that will be needed to evaluate the technical effort. Tools that can be used include:

- Primary project status meetings can be used to discuss the project and set the recommended frequency of the meetings.

- Technical project reviews can be held and should include:

  - System requirements review (SRR)
  - System design review (SDR)
  - Hardware requirements review (HRR)
  - Software requirements review (SWRR)
  - Preliminary design review (PDR)
  - Final design review (FDR)
  - Test readiness review (TRR)
  - Operational readiness review (ORR)
  - Intelligent transportation system project cost and schedule control system

- Weekly program status meeting can be held to address:
    - Key project issues
    - Responsible key project personnel
    - An example of a meeting agenda
    - Schedule changes
    - Current top 10 program risks
    - Risk mitigation effort status
    - Technical performance status
    - Upcoming significant events
    - New and old action items
    - Other issues

4.6.1.1 Formal Project Reviews

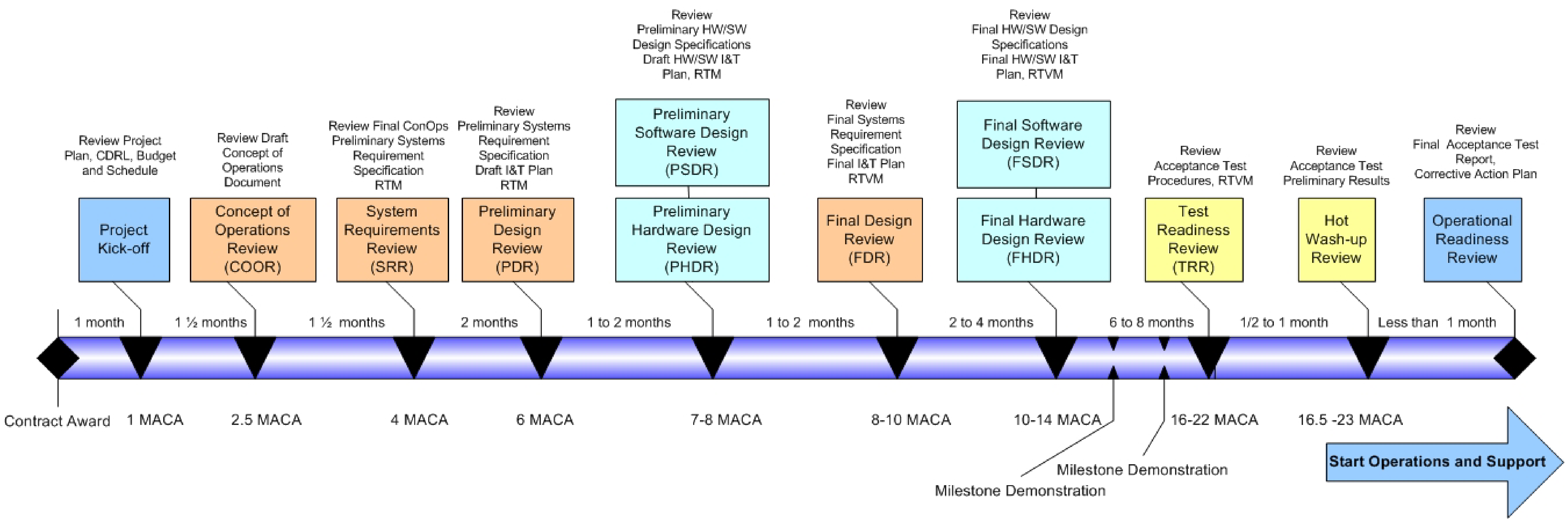
Formal reviews are held during the development of the project. Figure 4.8 shows the sequence of reviews that takes place during a normal ITS deployment project. The time line will vary depending on the size of the project, and whether hardware and software design reviews are needed.

4.6.1.1.1 Project Kickoff

It is presumed that the initial systems design work has been completed to arrive at a functional requirements specification based on user needs and the RITSA, and that a scope of services document has been published. These key documents form the basis of the procurement process that was discussed in detail in *Section 3*. The project reviews and time line can also apply to an ITS deployment that is not competitively bid; however, approximately three to six months must be added at the beginning of the project to accommodate the time for user needs and requirements analysis, and basic project planning.

At the kick-off meeting, the project team reviews the project plan that the ITS deployment team developed. The project plan spells out in detail how the scope of services' requirements will be satisfied and also provides an opportunity to finalize the process for managing the ITS deployment project with the contractor. It is more important to reach agreement on how the project will be managed and reported than to stick to a rigid formula for project management. Overall schedule, budget, and CWBS are reviewed, as well as the staffing plan. The CDRL should also be reviewed, along with any special format requirements. Formatting requirements are often spelled out in DIDs that accompany the CDRL.

Figure 4.8 – Formal Project Reviews and Typical Time Line



4.6.1.1.2 Concept of Operations Review

Before requirements are completely analyzed and finalized, the system design team should write a ConOps document that generally follows the *IEEE 1362-1998* standard.[60] Refer to *Chapter 5* for instructions on how to write this very important document. Briefly, the ConOps helps translate user needs into a common vision for how the system will operate in the user's environment.

The ConOps should provide the look and feel of the system for both the technical and non-technical ITS deployment stakeholders. After the draft ConOps has been reviewed and comments received, the concept is reviewed with the stakeholders. The product of the ConOps is consensus on a common vision for the system to be deployed. The ConOps will set the expectations, and have a strong influence on the design and implementation of the system. The ConOps must be done as soon as possible after user needs are determined. A typical concept of operations review (COOR) is about four hours and is usually presented using viewgraphs. Sometimes samples of similar hardware or software prototypes are demonstrated at the COOR to provide a better, more unambiguous look and feel to the system to be designed.

The ConOps may be created by the FDOT engineers or their representatives, or by a contractor after contract award. If the ConOps is done before the procurement phase,[61] the document should contain ROM[62] pricing on the hypothetical system described in the ConOps as well as a straw man schedule.

Before this review can be held, the following entry criteria must be completed:

- Completion of mission needs analysis
- Determination of stakeholder requirements for the ITS deployment
- Publication of the draft ConOps document

Products to be reviewed at the COOR include:

- Draft ConOps document
- Straw man schedule and budget, if completed before the procurement phase
- Hypothetical system architecture
- Potential risk items

---

[60] Institute of Electrical and Electronics Engineers, *IEEE 1362 – Guide for Information Technology – System Definition – Concept of Operations (ConOps) Document – Description* (1998).

[61] This refers to the procurement of a contractor to design and deploy the system according to FDOT specifications.

[62] A ROM is typically a gross estimate of the cost that is meant to indicate the upper limit of the expected cost.

The following COOR exit criteria must be completed before the project may proceed:

- Consensus reached on the vision for the system
- Consensus reached on the ROM budget and schedule

4.6.1.1.3 System Requirements Review

An SRR will be held to determine the functional baseline for the system (i.e., the ITS deployment). Before this review can be held, the following entry criteria must be completed:

- Completion of the ConOps document, with the final version approved
- Completion of the functional analysis, such that a necessary and sufficient set of system functional requirements has been defined

Products to be reviewed at the SRR include:

- ConOps document
- Draft system requirements specification
- Preliminary system architecture
- Requirements traceability matrix
- Risk items

The following exit criteria must be completed before the project may proceed:

- Consensus reached on the final version of the ConOps
- Consensus by all stakeholders on the functional system requirements baseline
- Consensus on the system architecture

4.6.1.1.4 Preliminary Design Review

This preliminary review of the system design emphasizes how the system design meets the system requirements. Typically, a PDR is held before the hardware design review (HDR) or the software design review (SWDR), since the hardware and software design responds to the system design requirements. It is recommended that the system design be presented in terms of what the system functions are and the major data flows between the functions. Emphasis at the systems level is *what* the system has to do, not *how* the system does it.

The PDR entry criteria includes:

- Completion of all action items from the SRR
- Completion of the system requirements allocation process
- Completion of a draft system requirements specification
- Completion of the HDR and the SWDR, if they were held
- Identification of all external system interfaces
- Completion of the functional and physical architectures
- Completion of preliminary supportability concepts

Products to be reviewed at the PDR include the:

- Draft system requirements specification;
- Preliminary SDD;
- Approved preliminary hardware design and software design documents, as appropriate
- Preliminary ICDs and ICS
- Functional architecture
- Physical architecture
- Risk items

Exit criteria for the PDR includes:

- Approved system design specification
- Approved SDD
- Preliminary agreement on external and internal interfaces
- Agreement on functional and physical architectures
- Agreement on basic supportability concepts

4.6.1.1.5 Hardware and Software Design Reviews

Further reviews are needed that address hardware- and software-specific issues. These reviews always follow the SDR and include the:

- **Hardware Design Reviews –** If hardware is being specified, and will be designed and fabricated by the FDOT or a contractor, one or more design reviews that focus specifically on the hardware is required. This review is needed only when hardware development is part of the ITS deployment; it is not often that the FDOT will specify special hardware requirements. Typically, the FDOT will purchase COTS hardware and software, and is more likely to specify unique software applications than hardware units. Custom-designed hardware may be needed if there are critical functions performed by hardware, and the interface of the hardware to the software in the system is particularly detailed or unique. If a lot of development work is required, a second review is held called the critical hardware design review (CHDR). The first review would then be the identified as the preliminary hardware design review (PHDR).

The entry criteria for the HDR includes:

- Completion of all action items from the SDR
- Completion of the system requirements allocation process to hardware functions
- Identification of all external and a preliminary set of internal system interfaces
- Completion of the functional and physical architectures
- Completion of the preliminary supportability concepts

Products to be reviewed at the HDR include:

- Complete updated system specification
- Preliminary hardware design document (HDD)
- Preliminary ICDs and ICS with an emphasis on the hardware/software interfaces
- Functional architecture
- Physical architecture
- Preliminary assembly drawings
- Risk items
- Test requirements, including environmental testing and certification

Exit criteria for the HDR includes:

- Approved hardware design specifications
- Approved HDD
- Preliminary agreement by software and systems engineering on the external and internal interfaces the hardware engineers propose
- Agreement on functional and physical architectures with an emphasis on command and control capabilities through software and software support for maintainability (i.e., built-in test [BIT] support)
- Agreement on basic supportability concepts
- Agreement on the certification requirements and process
- Agreement on the testing process, including environmental tests

- **Software Design Reviews –** Most often, the FDOT will procure custom software that controls commercial hardware, so one or more design reviews that focus specifically on the software is required. This review is needed only when software development is a part of the deployment. In general, if a project has a significant amount of deployment application software, such as with a TMC, a review is needed that concentrates on the software design with an emphases on how the design meets the system requirements. A single software design review on a project is called a SWDR to distinguish it from a SDR.

If a lot of development work is required, a second review is held called the critical software design review (CSDR). If the software will be designed, built, and deployed in stages, called releases, then it may be appropriate to have a SWDR for each software release. If a CSDR is held, then it must be preceded by a preliminary software design review (PSDR).

The entry criteria for the SDR includes:

- Completion of all action items from the SDR
- Completion of the system requirements allocation process to software functions and completion of the RTM
- Identification of all external and a preliminary set of internal system interfaces
- Completion of the functional and physical architectures
- Completion of preliminary I&T concepts
- Completion of the SDP

Products to be reviewed at the SDR include:

- Complete updated system specification
- Software development plan, if not reviewed earlier
- Review of the RTM
- Preliminary SDD with an explanation of what requirements are satisfied by the design and where they are satisfied
- Preliminary ICDs and ICS with an emphasis on hardware/software interfaces
- Functional architecture
- Physical architecture
- Risk items
- Test requirements, including the simulation capabilities and test bed requirements needed to support software testing

Exit criteria for the SDR includes:

- Approved software design specifications
- Approved SDD
- Preliminary agreement by hardware and systems engineering on the external and internal interfaces the hardware engineers propose
- Agreement on functional and physical architectures with an emphasis on command and control capabilities through software and software support for maintainability (i.e., BIT support)
- Agreement on the basic supportability concepts
- Agreement on the I&T approach
- Agreement on the testing process, including environmental tests

- **Firmware – The InBetween –** Often, the ITS engineer hears about firmware and may wonder what exactly it is. The term "hardware" is descriptive because the product can be readily seen, touched, and examined as it is being built and when it is delivered. Software is an equally descriptive term, though you cannot easily see or feel the product. You have to use other means to examine it, so it is hard to tell if the software development is making progress and is on schedule. This is the reasoning behind milestone demonstrations of software functionality during the development period.

  "Firmware" is software, but in a form that cannot be changed easily because it exists within a specialized hardware memory device. Examples of these are programmable read-only memory (PROM), erasable programmable read-only memory (EPROM), and electrically erasable programmable read-only memory (EEPROM). Firmware should be treated as software and reviewed as a software development effort would be.

4.6.1.1.6 Final Design Review

The final review of the system design is an important step in that it theoretically is the gateway leading to full-scale hardware and software development. In practice, hardware and software designs and implementations take place after their respective design reviews.

The entry criteria for the FDR includes:

- Completion of all action items from the PDR
- Completion of the system requirements allocation process;
- Completion of the final systems requirements specification document
- Completion of the HDR and SDR, if they were held
- Identification of all external system interfaces
- Review of the draft ICS
- Completion of the preliminary I&T concepts
- Completion of the functional and physical architectures
- Completion of system life cycle supportability concepts

Products to be reviewed at the FDR include:

- Final system requirements specification
- Final SDD
- Draft SAT plan
- Approved preliminary HDD and SDD, as appropriate
- Preliminary ICDs and ICS
- Functional architecture
- Physical architecture
- Risk items

Exit criteria for the FDR includes:

- Approved system design specification
- Approved SDD;
- Consensus on the acceptance test plan
- Agreement on external and internal (hardware/software) interfaces
- Agreement on functional and physical architectures
- Agreement on system life-cycle supportability concepts

4.6.1.1.7 Test Readiness Review

The TRR is a formal review conducted before starting a formal acceptance test of the system. It describes the objectives and contents of the review; when it should be held; and who should attend.

The entry criteria for the TRR includes:

- Completion of all action items from the FDR
- Completion of the final SAT plan
- Completion of preliminary acceptance test procedures
- Review of the preliminary ICS
- Completion of the final I&T plans[63]
- Completion of the draft system installation, and checkout plan and procedures
- Completion of the updated RTVM

Products to be reviewed at the TRR include:

- Final acceptance test plan
- Acceptance test procedures
- Preliminary installation and checkout plan
- Acceptance criteria and the process to correct deficiencies
- Updated ICDs and ICS
- Test support equipment needs
- Risk items

[63] Requirements for the milestone demonstrations should be specified in the I&T plan.

Exit criteria for the TRR includes:

- Approved final acceptance test plan
- Approved final acceptance test procedures document
- Approved preliminary installation and checkout plan
- Consensus on the pass/fail criteria and acceptance process
- Agreement on the process to correct deficiencies
- Agreement on the schedule for acceptance testing, including who needs to support and witness the testing

4.6.1.1.8 Hot Wash-Up Review

This review is held immediately after formal acceptance testing is concluded to obtain a consensus on the testing results and to discuss any major discrepancies. The hot wash-up review lays the foundation for the acceptance test report and the resulting corrective action plan, if one is needed. The meeting generally will not take more than an hour, and involves the contractor's project manager and test director; the FDOT ITS project manager; and any other key decision makers who have influence over the system's acceptance.

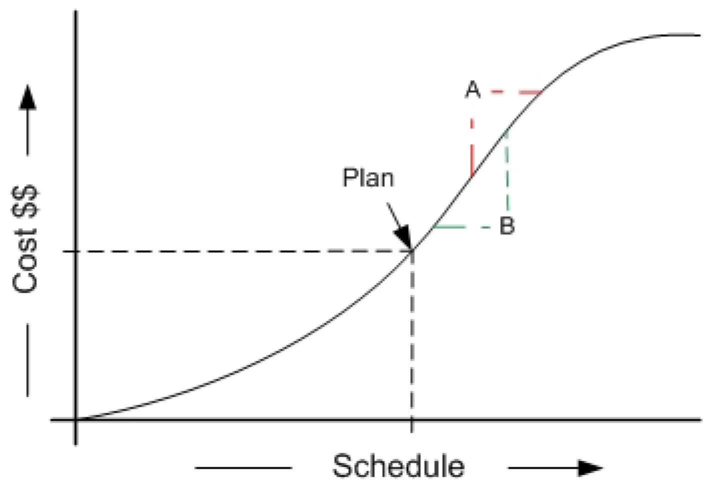
4.6.1.1.9 Operational Readiness Review

The ORR is held before full-scale deployment and operation of an ITS project. The ORR focuses on all the elements that need to be completed prior to operating the system. Topics range from training status to operations and maintenance procedures. The disposition of acceptance test discrepancies is reviewed and a final determination is made to proceed with the operation of the system while minor discrepancies are being corrected.

4.6.1.2 Cost and Schedule Monitor

This section describes how program performance is monitored with respect to cost and schedule. Cost and schedule are interrelated, and either one taken alone can be misleading. For a complete picture of a project's status, the value of the work performed to date must be evaluated. Work value is a product of both schedule and budget compared to the plan as shown in Figure 4.9. For example, is it acceptable to be ahead of schedule but to have spent more money to get there, as shown by the letters A and B in Figure 4.9?

Figure 4.9 – Work Value Example



The curve represents the project's planned cost expenditure over time. The project status indicated by the word Plan is right on budget and schedule. Now, assume that the project status was really indicated by the letter A, indicating that the project is ahead of schedule, which is good, but also that it exceeded the planned budget to reach that point in the schedule, which is bad. Therefore, the project has overspent to achieve that point in the schedule or it's behind schedule based on the amount of budget expended – neither of which is good.

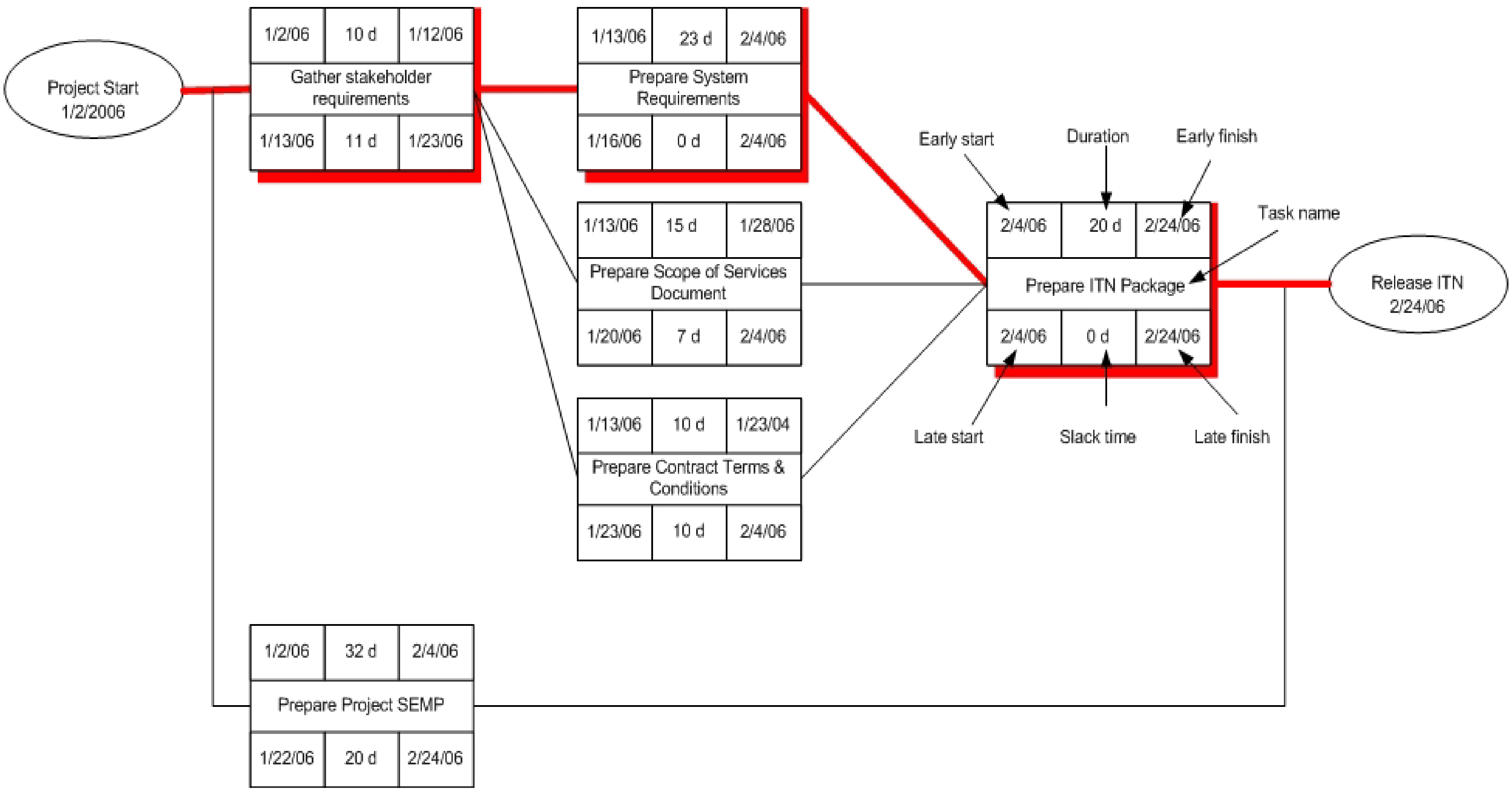
The letter B is an example of a project that is also reported ahead of schedule but over budget because the amount of additional budget spent has a far greater value in terms of work accomplished. Project status B is favorable and status A is not.

It is recommended that the budget and schedule plan be graphed and tracked in monthly status reports. The trend from month to month will indicate whether corrective action is effective for projects that are in trouble. To help the ITS project manager focus on what is important, it is recommended that the project's critical path be determined and the status of the critical path be monitored closely. This is done using the PERT chart.

4.6.1.2.1 Project Evaluation and Review Technique

The project schedule created by the contractor must be done using the network diagram view in Microsoft Project. Each task in the detailed work plan is entered and linked to one or more tasks that depend on it being completed before the task can be started. It is critical that these dependencies are identified accurately and that no task is left open-ended. An example of a closed network PERT is shown in Figure 4.10.

Figure 4.10 – Sample PERT Network



The critical path in the example PERT is indicated by the heavy lines and shadowed task boxes. It is the critical path because at least one task has zero slack time. It is suggested that those who are not familiar with PERT manually construct a simple network and work forward to the end, entering the early start dates and durations. Calculate the early finish dates and if more than one task connects to a single task to the right, the latest early finish date becomes the early start date for the task to the right.

Once the last task is reached, enter the earliest finish date as the latest finish date in the lower right-hand box and then work backwards through the network using the latest start date as the latest finish date in the preceding task box to the left. Once done, calculate the difference between the early start date and the latest start date to get slack time. The critical path through the network is the path that has one or more tasks that have the least slack time.

Figure 4.11 shows the recommended information that should be entered in the PERT task box. The PERT is created by sketching out the approximate task flow on a page and then using Microsoft Project to create the task box detail. The PERT network is first created assuming unlimited personnel are available (i.e., the task network is not resource-limited initially). The task name should relate to the detailed project plan's list of work tasks.

Figure 4.11 – Recommended PERT Task Box Information

<table>
<tr><td>Early Start</td><td>Duration</td><td>Early Finish</td></tr>
<tr><td colspan="3">Task Name</td></tr>
<tr><td>Late Start</td><td>Slack</td><td>Late Finish</td></tr>
</table>

- **Early Start – Duration – Early Finish –** The early start date is the earliest date that a task can start. The duration is typically entered in man-days, not calendar days. The early finish date is calculated based on the duration value entered.

  Work through the network creating the task boxes, and do not bother entering the late start dates or the slack. The software will calculate those values. The important point is to make sure all the tasks connect to others so that every task links to the project start milestone and the project end milestone. Also, identify any outside products that are needed by specific tasks. These outside products are created as milestones and are the only items that can have no preceding task driving them.

- **Project End Date Milestone –** When you connect the last task in the network to the project end milestone, set the latest date that you can permit the project to end. This date will cause the software to calculate the latest start and end dates; the slack time; and the critical path for the project.

- **Latest Start – Latest End – Slack –** If you can start a task on or before the latest start date, you will not change the project end date. If the duration of a task is longer than planned and the latest end date is exceeded, the end date of the project will slip and the task will become a part of the critical path. Slack time is the difference between the early start date for a task and the late start date for a task. You can start the task any time during the slack time and not change the project end date. The slack time provides the flexibility to plan the assignment of personnel to the task to optimize the staffing plan.

- **Critical Path –** The critical path is the path through the network where slack is zero or negative. If slack is negative for any task, the network must be rearranged so that there is no negative slack time. A project may be started with some tasks shown to have negative slack time if a plan is in effect to recover time on other tasks, such as planned overtime. In general, the workday for the PERT network should be Monday through Friday, excluding holidays, using the normal working hours in effect in the organization. By doing this, the project manager has the flexibility of catching up through overtime.

4.6.2 Quality Management Planning

The QM process refers to the overall management functions that determine and implement quality policy. The QM process, through an effective QA program, establishes a uniform management policy and implements an effective configuration control program. Therefore, the processes presented in this document are based on both the QM and CM processes, and are tailored to meet the guidance and recommendations outlined in *Florida's Statewide SEMP*; the *EIA-649*[64] standard; the *ISO 10007*[65] standard; and the *IS0 9000*[66] family of QM standards.

[64] Electronic Industries Alliance, *EIA-649 – National Consensus Standard for Configuration Management, Revision A* (October 2004). More information regarding EIA standards is available online at http://www.eia.org/.

[65] International Organization for Standardization, *ISO 10007 – Quality Management Systems – Guidelines for Configuration Management* (2003). More information regarding ISO standards is available online at http://www.iso.org/.

[66] International Organization for Standardization, *ISO 9000 – Family of International Quality Management Standards*, (2000).

Given the complex process of fabricating, assembling, testing, and integrating various ITS devices, QM has become a critical aspect in efficient and cost-effective ITS project implementation. In fact, not only must QM address the internal processes of the organization, it should also consider the quality systems of its suppliers and subcontractors.

Quality system planning is the first stage in setting up a QM system within an organization. It pertains to the identification and acquisition of resources, logistics, and manpower needed for the organization to define and achieve the required quality. Quality planning should include compatibility among the various aspects of the company's operations from start to finish, including product design, production, and product inspection/testing. It should also include definition of product and PSpecss. Identification of the necessary monitors, as well as inspection/verification stations at suitable points along the production process, should also be addressed as early as quality planning. Measurement capability requirements must likewise be defined during quality planning.[29]

If an organization chooses to use the *ISO 9000* family of standards to obtain QM certification, which is recognized internationally, then quality planning should ensure that the organization's quality system will eventually conform to all the requirements of the 20 *ISO 9000* elements. *ISO 9000* QM certification requires extensive audit of the organization's quality system. However, certification is not mandatory, nor is it necessary in order to benefit from its principles and processes when implementing applicable ISO elements in the deployment of ITS projects.

4.6.2.1 Quality Assurance / Quality Control

A QM plan should contain both a plan to implement QC, and a plan to monitor and verify that quality standards are being achieved. Quality control is the process whereby quality is engineered into the products being deployed through inspection, testing, and audits of documentation. Quality assurance is the process of verifying that the product has met the quality standards established by the QM plan. An IV&V group performing functional testing and inspections of the system provide an excellent QA process.

Refer to *Section 7.3* for more information on QM planning and to the FDOT's *Statewide QA Plan for ITS Deployments* as referenced previously.

[29] Copyright © 2001 SemiconFarEast.com.

4.6.3 Lessons Learned

No project ever follows its plan and the art of project management is the ability to effectively manage change. The successful ITS project team will always learn from past mistakes and never repeat them. The team will also have the processes in place to effectively monitor the project as it is developed and deployed; to detect trouble early; and to react quickly to correct problems. A characteristic of a well-managed ITS deployment is one where many small problems are identified, and solved quickly and effectively, rather than a project whose status appears normal until near the end of the schedule and budget, where major problems are identified.

The project team should maintain an anecdotal log of lessons learned as the project evolves, and at the conclusion of the project the ITS project manager should produce a "lessons learned" document for distribution to other ITS teams and key project leads.

4.7 Control Technical Baseline

Both *ANSI/EIA 649* and *ISO 10007* are standards intended to be used when establishing, performing, or evaluating CM processes in any industry, business enterprise, or governmental organization. A discipline popularized by its use in the acquisition of defense systems, CM is widely used for commercial products and services. When CM principles are applied using effective practices, return on investment is maximized and product life-cycle costs are reduced.

The CM process applies appropriate procedures and tools to establish and maintain consistency between the product and the product requirements or attributes defined in product configuration information. Implementing CM requires a balanced and consistent application of CM functions, principles, and practices throughout the product life cycle.

The CM process facilitates the orderly identification of product attributes, and provides control of product information and product changes used to improve capabilities; correct deficiencies; improve performance, reliability, or maintainability; extend product life; or reduce cost, risk or liability.

4.7.1 Manage Configurations

The CM process is the set of activities that identifies, documents, and controls configuration items (CIs) relevant to a particular product. It is used to provide consistency between product requirement, product configuration information, and product attributes. Therefore, when implementing a CM program, the process must focus on customer requirements for the product and should take into account the context in which CM will be performed. As previously stated, the purpose of CM is to establish and maintain the integrity of the product(s) of a project throughout the product's life cycle.

Once a product or project has been defined, the CM process begins. It is an integral part of project management, and is critical for adequately implementing and administering a QA program. The CM activities are generally one of two basic types: base lining or change control. Base lining involves the managerial agreement on the content of a system (i.e., product) or CI. Change control is the process of developing, coordinating, approving, and documenting changes to CIs.

4.7.1.1 Project Baselines and Their Purpose

A project is baselined to freeze changes in schedule, budget, and technical scope. It is impossible to deploy an ITS project if requirements keep changing. It is better to freeze the requirements; build the system; deploy according to schedule and budget; and then incorporate change. There are three opportunities for baselines in a project deployment's development cycle: the requirements baseline, the design baseline, and the as-built baseline.

- **Requirements Baseline –** A requirements baseline establishes the system problem the ITS deployment is intended to solve. The operational view (i.e., the ConOps document) describes how the system products serve the users. It establishes who operates and supports the system and its life cycle processes, and how well and under what conditions the system products are to be used. The functional view (i.e., the functional requirements specification) describes what the system products do to produce the desired behavior described in the operational view, and provides a description of the methodology used to develop the view and decision rationale. The ConOps helps clarify the system's functional requirements. When the SRR is held and the system's functional requirements are frozen – meaning there will be no more changes allowed in the baseline – the requirements baseline is established. The schedule and budget will remain fixed, and deviations to them will be monitored and reported.

- **Design Baseline** – The design view (i.e., the system design specification) describes the development design considerations of the system products, and establishes requirements for technologies and design interfaces among equipment, and among humans and equipment. The approved SDD that results from the FDR establishes the system's design baseline, the final HDR establishes the hardware design baseline, and the final SWDR establishes the software design baseline, if hardware and software designs are reviewed and approved separate from the system design. When the design is approved, the schedule and budget should be revised to incorporate any deviations from the requirements baseline. The project is then managed according to the revised schedule and budget.

- **As-Delivered Baseline** – The acceptance test establishes the as-delivered baseline. This baseline is established by reviewing all the project documentation through a functional configuration audit (FCA) and the acceptance test serves the purpose of the physical configuration audit (PCA). The schedule and budget should be changed to incorporate any previous deviations and changes, and then used to manage ongoing operations and support of the ITS deployment.

4.8 Configuration Management Planning

4.8.1 Overview

Florida's Statewide SEMP defines the interdisciplinary tasks that are required throughout a system's life cycle to transform customer needs, requirements, and constraints into an ITS solution. The CM process is a critical element of the back-end phase of the SEP that is essential to the verification and validation of all system requirements identified in the front-end systems engineering phase. This section is intended to improve FDOT management performance, and to provide guidance in the technical and administrative application of CM processes over the life cycle of ITS products, which may include hardware, computers, and software applications.

4.8.2 Purpose

This section provides an overview of the standard CM processes essential for incorporation in the FDOT SEP (e.g., background and guidance), and to cite applicable standards and references to support the standard selected. This document:

- Defines the subprocesses and practices that comprise the CM approach
- Defines a set of standard practices and technical tools for use with the CM approach
- Establishes a formal process for implementing and improving configuration control and management functions

4.8.3 Configuration Management References

The CM process, which is a discipline popularized by its use in the acquisition of defense systems, is widely used for commercial products, systems, and services. The *EIA-649* standard was written to replace *Military Standard (MIL-STD) 973*[68] for use commercially and is recognized as the *National Consensus Standard for Configuration Management*. It defines CM terminology and establishes a CM process using five CM functions.

[68] United States Department of Defense, *MIL-STD-973 – Configuration Management* (September 2000).

Similarly, *ISO 10007* promotes a common understanding of CM and provides general guidance on the use of CM within an organization that can easily be mapped to the principles in the *EIA-649*, *EIA-632*,[69] and *IEEE 1220-1998* standards.

The *EIA-649* and *ISO 10007* standards are not intended for use as compliance documents for CM programs. However, they are intended for use as source and reference documents, much in the same manner as this section is intended. The appropriate application of CM functions and principles will enable the FDOT District ITS engineer, project manager, or project engineer to plan and implement a CM program for a product.

The term "product" in this document, as well as in the related standards, should be interpreted as applicable to the generic product categories, such as documentation, facilities, firmware, hardware, software, tools, materials, processes, services, or systems.

4.8.4 The FDOT Change Management Board

The FDOT is embarking on a program to accelerate the deployment of ITS through the expenditure of over $500 million in statewide-managed funds. These funds have been programmed to deploy ITS on the five principle limited-access facilities in the state over a 10-year period, with additional funds added each year to keep the 10-year plan horizon. This wide-spread deployment of ITS projects will be more cost effective if changes can be managed on a statewide basis.

Change is an inevitable consequence with the implementation of complex systems such as ITS. Technology is always changing and what was state-of-the-art yesterday will be obsolete tomorrow. To assure that change is implemented consistently from District to District, a process needs to be put in place to assure that change is applied throughout the state.

The purpose of the FDOT CMB is to implement change in a controlled process so that change is not implemented without regard to how the change will affect statewide systems.

The *SEMP's* CM process relies on the FDOT CMB to review and approve changes to approved project baselines.

[69] Electronic Industries Alliance, *EIA-632 – Processes for Engineering a System* (September 2003).

4.9 Configuration Management Principles

Both *EIA-649* and *ISO 10007* are standards that are intended to be used when establishing, performing, or evaluating CM processes in any industry, business enterprise, or governmental organization. A discipline popularized by its use in the acquisition of defense systems, CM is widely used for commercial products and services. When CM principles are applied using effective practices, return on investment is maximized and product life-cycle costs are reduced.

The CM plan applies appropriate processes and tools to establish and maintain consistency between the product, and the product requirements and attributes defined in product configuration information. The implementation of CM requires a balanced and consistent implementation of CM functions, principles, and practices throughout the product life cycle.

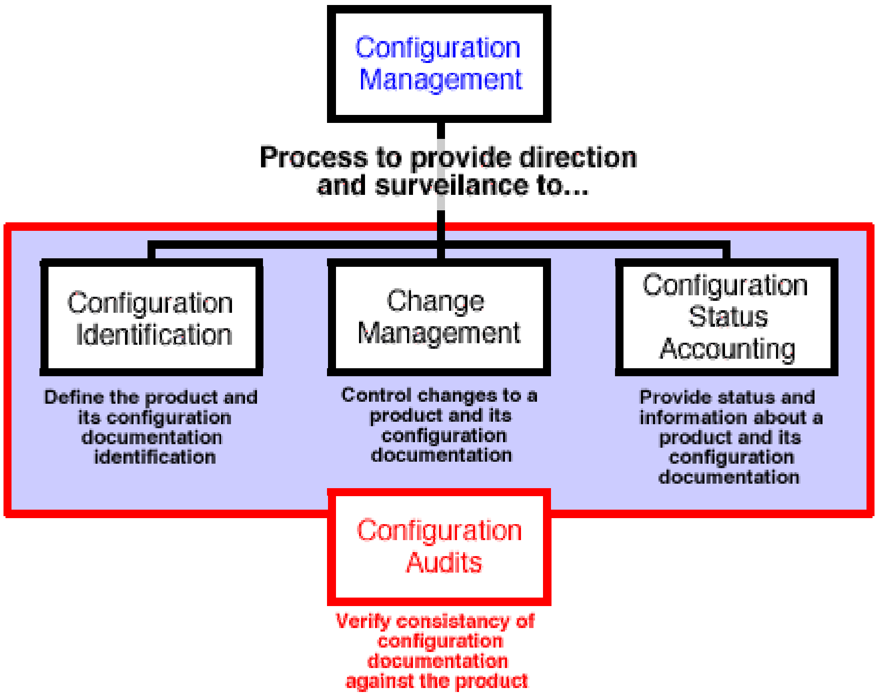
The CM process facilitates the orderly identification of product attributes and provides control of product information and product changes used to improve capabilities; correct deficiencies; improve performance, reliability, or maintainability; extend product life; or reduce cost, risk or liability.

This document defines CM terminology and establishes a CM process using the five CM functions outlined in the *EIA-649* and *ISO 1007* standards represented in Figure 3.11 and outlined below:

- Configuration management planning and management
- Configuration identification
- Configuration change control
- Configuration status accounting
- Configuration verification and audit

The CM block in Figure 4.12 includes the fifth function, CM planning and management.

Figure 4.12 – Configuration Management Functions[70]



4.10 Configuration Management

The CM process is the set of activities that identifies, documents, and controls CIs relevant to a particular product. It is used to provide consistency between product requirements, product configuration information, and product attributes. Therefore, when implementing a CM program, the process must focus on customer requirements for the product and should take into account the context in which it will be performed. As previously stated, the purpose of CM is to establish and maintain the integrity of the product(s) of a project throughout the product's life cycle.

Once a product or project has been defined, the CM process begins. It is an integral part of project management, and is critical for adequately implementing and administering a QA program. The CM activities are generally two basic types: baselining and change control. Baselining involves the managerial agreement on the content of a system (i.e., product) or CI. Change control is the process of developing, coordinating, approving, and documenting changes to CIs.

[70] © CMstat™ Corporation, San Diego, California. CMstat is a trademark of CMstat Corporation.

The implementation of a successful CM program begins with the identification of responsibilities and authority related to the implementation and verification of the CM process.

4.10.1 Responsibilities

It is important for the individual with overall responsibility for the product to develop a project organization, if not done previously, and assign CM responsibility to key personnel. In most cases, the FDOT District ITS engineer or project manager will be responsible for developing the project organization.

The following assignments of responsibility address QA and CM principles. The duties and responsibilities for CM at the product level shall be specified in appropriate CM documentation.

4.10.1.1 District ITS Engineer

The District ITS engineer has overall responsibility for the District ITS CM program and plan, and for providing guidance for the various products to ensure a successful development and implementation throughout the products/projects life cycle, and to establish the appropriate processes and systems to ensure compliance with the District CM plan. The District ITS engineer has oversight responsibility to ensure that a specific CM plan, tailored for a given product, is developed according to the District CM plan.

4.10.1.2 Project Manager or Project Engineer

The project manager or project engineer is delegated with the overall operational authority to implement the product's CM program according to the FDOT's ITS CM program. The project manager, project engineer, or specified designee may execute this authority. The project manager, project engineer, or designee shall:

- Develop the CM plan, tailored for a given product according to the District CM plan.
- Resolve CM program issues that affect multiple projects or subprojects within a program.
- Serve as the District's primary interface with the contractor (i.e., client) or the FDOT's TEOO ITS Section.
- Arrange for and provide independent evaluations of the product's CM program and an assessment of its implementation. The independent evaluation may be conducted under the direction of the ITS Section.

4.10.1.3 Assurance Review Office

The project manager, or project engineer, responsible for a product's CM will coordinate with the FDOT ITS Section in Tallahassee to schedule periodic assessments of the program. The FDOT ITS Section shall perform the functions of the assurance review office (ARO) to provide independent periodic assessments of the effectiveness of a product's CM program as defined in this document, the project-specific CM plan, and the status of implementation within the project.

Individuals knowledgeable in the CM field shall be assigned to conduct the assessments. It is recommended that the independent assessments review the five CM elements and the project specific requirements in the CM plan at least three times during a 12-month period. The physical configuration, document integrity, software integrity, and CM program consistency shall be evaluated, and the results of the independent assessments documented and reported to the District ITS engineer.

4.10.1.4 Configuration Management Subject-Matter Expert

The District ITS engineer or the project manager shall designate the CM subject-matter expert (SME) as the individual who is technically competent in the CM subject-matter area. The CM SME shall:

- Review changes in CM standards and CM-related directives.
- Recommend and author changes to a product's CIs.
- Recommend changes to the District CM plan;
- Advise the District ITS engineer, project manager, or project engineer on CM implementation issues.
- Provide for CM training, if required.

4.10.1.5 Configuration Item Owner

The CI owner is the SME for the CI and may be designated by the District ITS engineer or project manager. The CI owner shall:

- Identify system requirements, performance criteria, and documentation considered to be essential to system operation, if a product's design basis has not been previously defined.
- Maintain the configuration of the CI.

- Initiate the configuration change process.
- Determine the required documentation (e.g., drawings, calculations, specifications, test criteria, and vendor manuals) that defines the CI design requirements.
- Ensure the maintenance of system (i.e., product) documentation, records, and software, and keep all up to date.

4.10.2 Disposition Authority

The dispositioning authority, the CMB, is a decision-making entity that examines and decides whether changes to requirements, product design, etc., are cost effective and whether they should be pursued. The authority may be a person or a group of persons, but usually consists of a group of persons whose membership consists of subject matter, fiscal, and project (i.e., product) expertise capable of evaluating the proposed change. They have the responsibility and authority to make decisions on the interrelated functional and physical characteristics of a product's characteristics or configuration information.

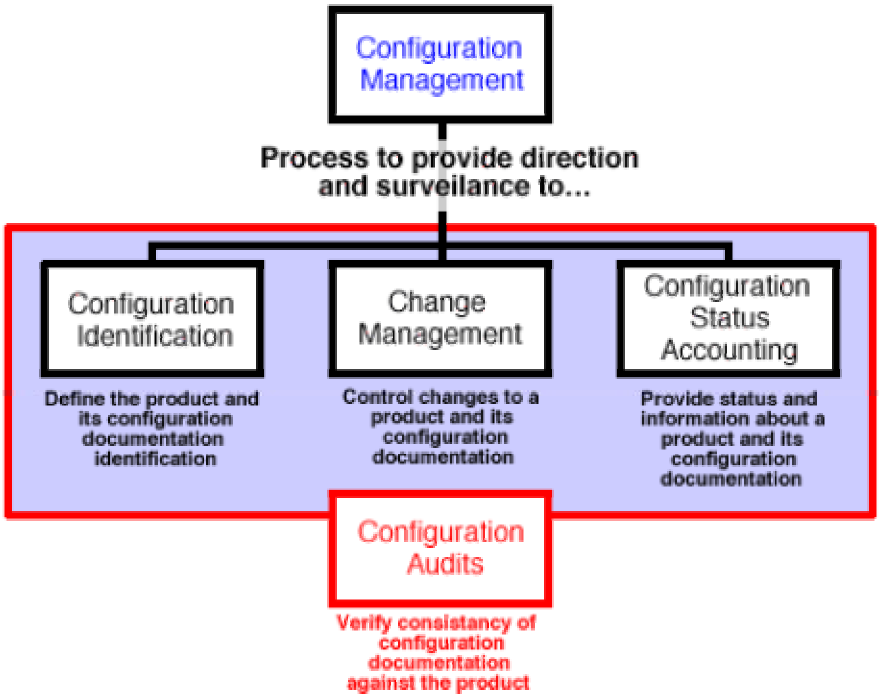
4.11 Configuration Management Process

The implementation of the FDOT's CM program is fundamentally based on the *ISO 10007:2003(E)* format and structure. *Appendix M* is an informative example of the structure and content of a CM plan extracted from the *ISO 10007:2003(E)* standard. The format of the paragraphs in this section follows the format presented in the *ISO 10007:2003(E)* standard.

4.11.1 General

This document defines CM terminology and establishes a CM process using the five CM functions outlined in the *EIA-649* and *ISO 10007* standards and represented in Figure 4.13. The CM block in Figure 4.13 includes the fifth function, CM planning and management.

Figure 4.13 – Configuration Management Functions



4.11.2 Configuration Management Planning

The foundation for the CM process is CM. As depicted in Figure 4.13, the management and planning elements of the CM process provide the structure for implementing CM in much the same way as the program management plan effectively manages a program. The objective of the CM planning element is to direct and monitor the development and implementation of the overall CM program. The purpose of CM program management is to provide processes that define program objectives, and to identify the actions and tasks for accomplishing and managing those objectives. The CM process should be detailed in a CM plan, and should describe any project-specific procedures and extend their application during a product's life cycle. (Refer to *Appendix M* for a template.)

As with program management, the CM program management functions may also include estimating the level of effort needed to complete each task; organizing and scheduling the planned tasks; staffing an organization to accomplish the planned tasks; assigning personnel to specific tasks; monitoring progress during implementation; identifying problems and taking corrective actions; and recognizing tasks and program completion.

The required activities associated with CM planning are:

- Identification of personnel in project organization with CM responsibilities
- Development of the CM plan and other CM documentation that may be applicable
- Establishment of controls for the CM plan
- Determination of the CIs based on product requirements and functionality
- Determination of the configuration procedures and the owner of each CI
- Identifying the controlling documentation for each CI
- Control of CI changes
- Evaluation of the effectiveness of the CM program's implementation
- Description of the CM responsibilities throughout a product's life cycle

The *ISO 10007:2003(E)* standard suggests that a product's CM plan may be implemented as a stand-alone document or as an integral part of another document. It is suggested that the District ITS engineer implement stand-alone, product CM plans for easy reference and traceability by all product stakeholders, such as clients, customers, the FDOT ITS Section, District ITS offices, product engineers, etc.

4.11.3 Configuration Identification

The basic unit of CM is the CI, which is determined by the configuration identification process. Typically, the most important elements of the product or system are identified as a CI. Configuration identification defines a product's structure and component-associated design requirements; physical configuration; documentation; and software characteristics. As a whole, this aggregate constitutes and defines a CI. Configuration identification establishes:

- A method for organizing the composition of product elements and associated information

- Unique identification of products and product configuration information

- Consistency between a product itself and the information about the product

- Product attributes that are defined, documented, and baselined

4.11.3.1 Identifying the Design Document Set

Each CI shall have a designated set of documentation that defines the design requirements and corresponding design basis. The CI owner is responsible for determining the specific documentation that the CI design requirements shall consist of.

The documentation that defines product/system design, including the acquisition program baseline, requirements documentation, system specifications, and the product baseline are elements of the CIs, and are changed as the hardware and software items are changed. Depending on the complexity of the product/system and their interrelationships, the specific CI documentation may not be the same for every CI.

4.11.3.2 Determining the Adequacy of Design Requirements

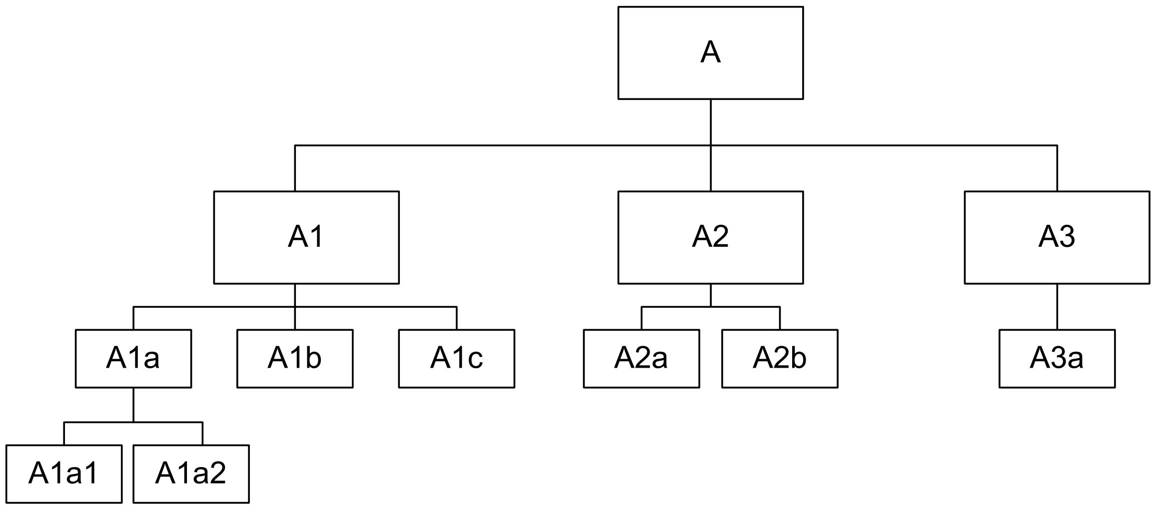
An appropriate level of technical management review should be held to validate and document the set of design requirements that are to be maintained for each CI. Where design requirements are identified, but are not going to be controlled as CIs, the reason for the decision should be documented. An initial assessment is one mechanism that can be used during the implementation of CM to determine the adequacy of design requirements. Other mechanisms include operational experience; maintenance and operational tests; and surveillance tests.

Grouping a set of design requirements essentially defines the functional and physical characteristics of a CI. Product configuration information should be relevant and traceable. A common technique for organizing product configuration information is to develop a product structure. It provides a basis on which to relate products, component products, and information by showing the top-down relationships among the various parts that make up the product and the quantity of each. A sample product structure is illustrated in Figure 4.14. A product structure is useful in visualizing the relationships, in determining the level(s) at which to apply CM, and in evaluating the impact(s) of proposed changes to the product. A unique product identifier, such as the "A1a" alphanumeric character string shown in Figure 4.14, is assigned for each product. This ensures that one product can be distinguished from another; one configuration of a product can be distinguished from another; units of the product can be distinguished from other units of the product; and the source of a product can be determined. Not only is a CI uniquely identified, but also, as a result, it provides a detailed record of requirement and change traceability of all product and sub-product changes.

4.11.3.3 Description and Identification of Configuration Items

Information related to product configuration is uniquely identified and linked to the specific product identification so that it can be referred to precisely and retrieved when necessary. To be unique, product configuration identification includes an identifier and the source of the identifier, as well as a description with a revision or version identifier so that the relationship to the product baseline can be maintained.

Figure 4.14 – Example Product Structure



The following information should be documented or referenced in the appropriate CM documentation:

- Unit identifier, which remains the same throughout life cycle
- Title or name of the CI
- Configuration management level (i.e., the hierarchy in the product structure)
- Version and date
- Configuration item owner
- Brief description of the CI
- Required CI documentation, records, and software components (e.g., safety basis documents; specifications; design documents; design criteria; performance criteria; formal basis of design documents; operations and maintenance manuals; lists; and software data)
- The functional organizational structure of those involved in the CM process

4.11.4 Change Control

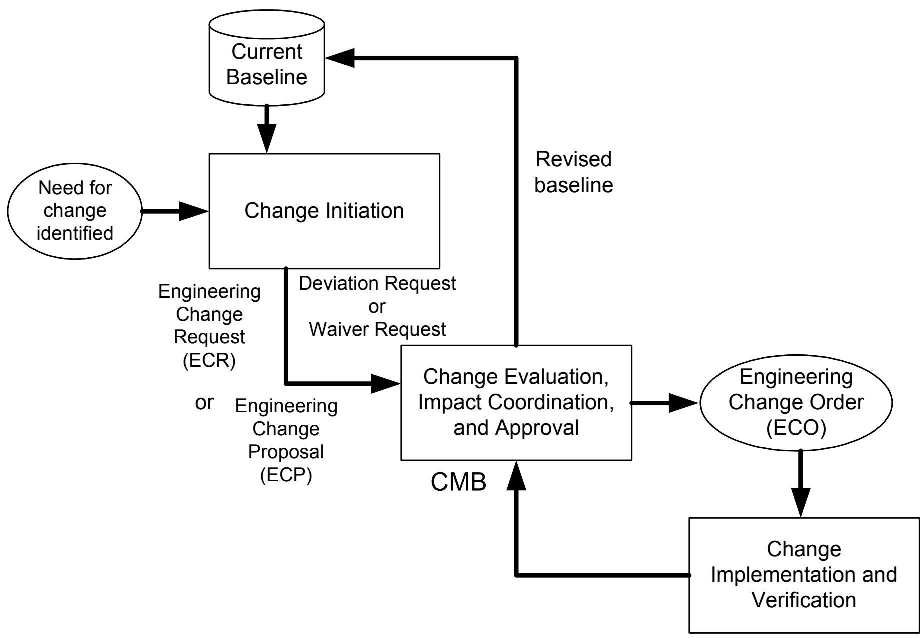
Generally, the terms of a contract will specify that once the design requirements baseline is determined, no changes will be made to the baseline. Once a product baseline configuration has been established, all changes must be controlled and documented. New requirements will not be incorporated into the baseline unless included with an authorized formal contract amendment.

Technical, budgetary, and scheduling problems for a CI must be diagnosed as early as possible to determine their impact. The objective of the change control element of CM is to maintain consistency and traceability of the design requirements, physical configuration, and changes made to documentation. This objective can be met if the following is used.

- The need for a change should be identified, and a CI change proposal initiated by describing the change(s) that needs to be considered and the justification for the change(s).
- The impact of the change, if approved or disapproved, should be evaluated and documented.
- A recommendation of how the change should be considered should be provided.
- A recommendation of how the change will be implemented and verified should be provided.
- Any procedures, documentation, and instructions required for incorporating the approved change in the product, as well as its related product configuration information, should be identified.
- Change incorporation and continued consistency with product configuration information that needs to be updated as a result of the change should be verified, and a schedule for completion and verification should be proposed.
- Approvals and variances from the baselined product requirements should be identified and documented when implemented.
- All change processing, including decisions, must be documented and maintained as a CM item.
- Notification of change approval or disapproval should be published and provided to all affected parties.

The configuration change management process model shown in Figure 4.15 illustrates the typical flow of the change control process. It is capable of addressing permanent changes, such as engineering change requests (ECRs) or engineering change proposals (ECPs), to provide a new product configuration, as well as temporary departures, such as requests for deviation or waiver, from the approved configuration that will allow the delivery of a nonconforming unit.

Figure 4.15 – Configuration Change Management Process Model



4.11.4.1 Identification of Changes

At the heart of an effective CM program is the currency and accuracy of the physical configuration of a CI. Changes may include changes to hardware, maintenance procedures, processes, operations, documentation, computer software, and inventory limits, as well as temporary modifications.

The FDOT; a customer; or a supplier or contractor may initiate a change. All change proposals must be uniquely identified and documented.

4.11.4.1.1 Engineering Change Requests or Engineering Change Proposals

To ensure that a change is both necessary and adds value, it is important to establish criteria for initiating requests for change or variance. Technical, budgetary, and scheduling problems must be detected and diagnosed as early as possible to determine their impacts, and whether or not an ECR or ECP is warranted. An ECR identifies a change and the CIs that are expected to be affected.

The difference between an ECR and an ECP is the amount of detail. An ECP will be a more formal document with justification for the change, the benefit to be realized, and the cost of the change. Typically, an ECP is initiated by a contractor in response to an ECR. An ECR can be generated by anyone on the project, once the procuring agency agrees that a change is necessary. An ECP should be prepared by the prime contractor, and submitted to the FDOT and the CMB for review and approval. The approved change becomes the engineering change order (ECO) that is issued to the contractor to effect the change. An ECO also modifies the contract when it is incorporated in the contractual documentation governing a project.

It is important to differentiate between actual changes, such as modifications and alterations, and maintenance work that does not affect the configuration but that does require monitoring, control, and status accounting. The basic relationship between design requirements, documentation, and physical configuration should not change because of maintenance work, whereas modification work entails configuration changes.

Configuration change management, as applied to change initiation, consists of the following subordinate processes:.

- Conceptually visualize the change to evaluate its benefits, such as the implementation of safety, quality, cost, produce-ability, and performance improvement measures, or to correct deficiencies in the product.
- Determine one or more approaches to accomplish the requested change.
- Define and evaluate each approach's impact to the product.
- Conduct preliminary assessments of the effect of making, or not making, the change.
- Classify the change to define the required levels of processing and approval authority, including customer approval or concurrence, when required.

Although a change may be initiated by the FDOT, a customer, or a supplier, the change initiation must be processed by the CI owner prior to submitting the ECR to the CMB. It is critical that the CI owner review the ECR or ECP, and identify all connected or impacted systems or documentation that may be affected when evaluating a change for submittal.

4.11.4.1.2 Information to be Included in Change Proposals

The following information provided by the *ISO 10007:2003(E)* standard should be included in change proposals:

- Configuration items and related information to be changed, including details of their title(s) and current revision status
- Description of the proposed change(s)
- Details of other CIs or information that may be affected by the change
- Interested party preparing the proposal and the date it was prepared
- Reason for the change
- Category of the change

4.11.4.2 Technical Review of Changes

Evaluations of the proposed change must be performed and documented. The extent of the evaluation should be based on the complexity of the product and should include:

- Technical merits of the proposed change
- Risks associated with the change
- Potential impact on the contract, schedule, and cost

Each proposed change must be reviewed to determine whether it is within the bounds of the design requirements. If not controlled, operational activities could result in unintended and undocumented CI changes. Changing equipment set points for operator convenience; lifting leads; using mechanical or electrical jumpers; pulling circuit cards; disabling enunciator alarms; and making computer software changes are all examples of operational activities that could have secondary effects on the original configuration. Such operational activities, if applicable, must be evaluated before implementation to ensure that the activities do not go undocumented or deviate from the established design requirements.

4.11.4.3 Management Review of Changes

The authority required to make a decision on changes varies with the magnitude and complexity of the change concerned. This decision-making process is an integral part of the overall project management of a project's systems engineering and CM processes. To enable the processes to operate effectively, it is necessary to establish specific procedures for dealing with, evaluating, and implementing changes according to the established rules and the formally constituted CMB.

The CI owner will ensure that a technical evaluation of the ECR has been conducted and that all of the required ECR/ECP information identified in *Section 4.4.1.2* has been assembled for submittal to the CMB. The CI owner will ensure that all members of the CMB have been provided with the ECR/ECP package, and that a CMB review meeting has been scheduled to determine the disposition of the ECR. If the ECR is approved, the CI owner will generate a formal ECP that contains the detail necessary for a complete evaluation and approval.

With the supporting documentation provided by the ECP, along with supplemental information provided at the CMB review meeting, the CMB will determine the disposition of the ECP with respect to the:

- Technical merits of the proposed change
- Risks associated with the change
- Potential impact on the contract, schedule, and cost

The decision of the CMB must be documented and recorded as part of the CI. Notice of the decision must be provided to relevant parties.

4.11.4.4 Implementation and Verification of Change

The ECO is the approval for the ECP and is a detailed specification of the changes needed. The implementation of an approved change includes:

- Changes to the product configuration information
- Notification of changes released to all concerned
- Verification of compliance with the approved changes

Before implementation of an approved change, as part of the change control process, management and the CI owner should review the change, even if it is not a change to the design requirements, to verify that the technical reviews have been performed adequately; the change package is complete and ready for implementation; and any necessary external approvals have been obtained prior to implementation.

As appropriate, postinstallation tests may be performed and must be evaluated for compliance with the modified CI and the established baseline. A new CI may require a system test ensure proper integration with other systems. Initial installation testing may be accomplished by QA acceptance testing.

4.11.5 Configuration Status Accounting

Configuration status accounting (CSA) is the function that provides an accurate, timely information base concerning a product and its associated product configuration information throughout the product's life cycle.

Configuration status accounting activities should be performed throughout the product's life cycle in order to document the evolution of a product from the initial baseline to its current configuration. It also serves as an indication of the effectiveness and efficiency of the CM process implemented for the product.

The following two paragraphs, extracted from the *ISO 10007:2003(E)* standard, provide general guidelines for maintaining records and reports, and are recommended for inclusion in the CM plans.

4.11.5.1 Records

During the configuration identification and change control activities, CSA records will be created. These records allow visibility and traceability, and the efficient management of the evolving configuration. They typically include details of the:

- Product configuration information, such as the identification number, title, effective dates, revision status, change history, and its inclusion in any baseline. Evolving product configuration information should be recorded in a manner that identifies the cross-references and interrelationships needed to provide the required reports.

- Product configuration, such as part numbers, product design, or build status

- Status of the release of new product configuration information

- Processing of changes. All ECR decisions (i.e., approvals or rejections) should be recorded and referenced to the CI.

4.11.5.2 Reports

Reports of varying types will be needed for CM purposes. Such reports may cover individual CIs or the complete product. Typical reports include:

- A list of product configuration information included in a specific configuration baseline
- A list of CIs and their configuration baselines
- Details of the current revision status and change history
- Status reports on changes and concessions
- Status details of delivered and maintained products concerning part and traceability numbers and their revision status

4.11.6 Configuration Audit

Normally, there are two types of configuration audits. Their definitions, as defined by the *ISO 10007:2003(E)* and *EIA-649* standards, are as follows:

- **Functional Configuration Audit (FCA) –** This is a formal examination to verify that a CI has achieved the functional and performance characteristics specified in its product configuration information; and
- **Physical Configuration Audit (PCA) –** This is a formal examination to verify that a CI has achieved the physical characteristics specified in its product configuration information.

Both may be required before formal acceptance of a CI and, if required, will be documented as part of the contract. It may not be necessary for all CIs to undergo a formal configuration audit, but the CM plan should clearly state what criteria will be utilized in verifying compliance with the requirements and approved changes.

Less formal, yet extremely important in CM effectiveness, are audits, or self assessments, conducted periodically throughout the product's life. The objective of such assessments is to help define CM needs, and to measure the effectiveness of the CM program in establishing and maintaining a product's basic relationships.

The District ITS engineer shall retain responsibility for the conduct of CM assessments and shall ensure that the CI owner will implement the corrective actions for deficiencies noted in the assessments.

4.12 Measurement

There are two types of project measurement tools: program metrics and technical performance measures (TPMs).

4.12.1 Program Metrics

Program metrics indicate the health of the program in specific areas, such as cost, schedule, budget, etc.

4.12.2 Technical Performance Measures

The TPMs are usually derived as part of the risk management process. They help control critical product parameters, and are usually a physical or logical product-focused measure. The list of metrics should include thresholds, corrective actions, status reporting, and review processes. The status of TPMs should be part of the program status reviews.

The TPMs must be chosen that are unambiguously verifiable at key points in the project development cycle. Milestone demonstrations of important functions are the recommended way to do this. Refer to *Chapter 3*, *Section 3.2*, for more information on TPMs.

4.13 Trade Studies

The term "trade studies" is an abbreviated way of saying trade-off studies. Trade-off studies are cost/benefit analyses used to help the project team decide on the best and most cost-effective technical solution. There can be a number of trade studies on a project to weigh the merits of different technologies and deployment options.

4.14 The FDOT Statewide SEMP Support Plans and Practices

The life cycle of any system or its major components consists of a sequence of activities, including design, development, production, deployment (i.e., testing, checkout, and training), support, operation, and disposition. Systems engineering activities span the entire system life cycle from systems analysis, requirements definition, and conceptual design at the outset of a project through production, operational support, planning for replacement, and eventual retirement and disposal at the end.

The SEP is an iterative process of technical management, acquisition and supply, system design, product realization, and technical evaluation at each level of the system, beginning at the top (i.e., the system level) and propagating those processes through a series of steps that eventually lead to a preferred system solution. At each successive level, there are supporting processes necessary to manage and control the project activities. Systems engineering involves planning for different aspects of the project that eventually evolve into a set of plans for managing and controlling these processes.

This section describes different processes in a system's life cycle and their associated support plans. The description is intended to provide ITS engineers with a basic understanding of each process/plan so that they can select the ones that are appropriate to the ITS project under consideration. Complimentary to the process/plan description, the appendices contain a synoptic outline in a standard format for each plan. These include templates with instructions so the ITS engineer can readily use them to develop the subject support documents.

A typical set of process support plans is listed below. Many of these plans are directly applicable to a majority of ITS (or engineering) projects. Some of the plans fall under the category of general project management and control. A significant number of these support plans belongs to the area of engineering specialties. The engineering specialty plans will be discussed in *Chapter 6*. The general support plans include:

- Program management plan
- Operational development plan
- System/Subsystem requirement specifications
- Software development plan
- Hardware development plan
- System integration documentation, test plans, procedures, and reports
- Configuration and data management plan (CDMP)
- Quality management plan
- Quantitative management plan
- Subcontract management plan

4.14.1 Description of General Support Plans

4.14.1.1 Appendix E – Program Management Plan

The program management plan (PMP) establishes the management approach used on an FDOT ITS project that is consistent with the approach used on all FDOT ITS programs. The PMP describes the overall program structure; deliverables; related management plans and procedures; and the methods used to plan, monitor, control, and improve the project development efforts. The PMP is a dynamic document, and is updated on a periodic basis to reflect all organizational changes, lessons learned, and advances in methodologies that occur throughout a project's life cycle.

4.14.1.2 Appendix F – Operational Development Plan

The operational development plan (ODP) describes the necessary tasks, responsibilities, and controls that the FDOT and ITS project subcontractors will implement. Its primary objective is to assure that the FDOT ITS project has sufficient and significant resources to support the project objectives. The ODP provides sufficient details to cover the top-level operating concepts. The details of operation should be defined during the subsequent analysis, design, and development phases. The OPD usually covers the following elements:

- Project summary
- Description of the overall mission of the system
- Description of the overall system requirements
- System milestones
- Defining the current project resources

- Constraints and risks
- Future system improvement efforts
- Supporting plans and work instructions

4.14.1.3 Appendix G – System / Subsystem Requirements Specification

The system/subsystem requirements specification establishes the functional, performance, design, development, and verification requirements for an ITS project. This document contains requirements gathered from the contractual elements typically consisting of the contract documentation, SOW, operational requirements document, RFP, etc. The system/subsystem requirements specification defines the set of requirements to be verified as part an ITS project's acceptance.

4.14.1.4 Appendix H – Software Development Plan

The SDP establishes the software development approach, methodologies, tools, and procedures to be used during the analysis, design, development, testing, integration, deployment, and maintenance of the software for each FDOT ITS project. The SDP is a dynamic document and shall be updated on a periodic basis to reflect organizational changes, lessons learned, new tools, and advances in methodologies. The SDP would not be directly applicable to the FDOT ITS engineer, since typical FDOT activities do not usually include software development. However, the SDP should be a requirement for project subcontractors responsible for developing and submitting the SDP document for a software development effort.

4.14.1.5 Appendix I – Hardware Development Plan

The HDP establishes the hardware development approach, methodologies, tools, and procedures to be used during the analysis, design, development, testing, integration, deployment, and maintenance of the hardware for each FDOT ITS project. The HDP is a dynamic document and shall be updated on a periodic basis to reflect organizational changes, lessons learned, new tools, and advances in methodologies. The HDP would not be directly applicable to the FDOT ITS engineer, since typical FDOT activities do not usually include hardware development. However, the HDP should be a requirement for project subcontractors responsible for developing and submitting the HDP document for a hardware development effort.

4.14.1.6 Appendix J – System Test Plan

The system test plan (STP) establishes the methods needed to verify that system end-items satisfy their requirements. The STP addresses verification requirements and criteria for solution alternatives; the definition of verifications to demonstrate proof of concept; and development, qualification, acceptance, and pertinent operational and other testing.

4.14.1.7 Appendix K – System Test Procedures

The system test procedures prescribe the procedures to be followed to verify each requirement the system must meet. The purpose of the test procedures document is to provide a repeatable series of steps that will result in the system producing the same result. The test procedures also serve as the data sheets for when the test is run and become a part of the test report.

4.14.1.8 Appendix L – System Test Report

The test report summarizes testing results, explains what failed and why. The test report often serves as the official document to accept the system, so it must be absolutely accurate. The test data sheets should be included exactly as they were used with pen and ink changes if necessary.

4.14.1.9 Appendix M – Configuration and Data Management Plan

The CDMP establishes those methods and practices to be employed in the identification, change control, status accounting, and audit of FDOT ITS projects. The plan describes the CM process to be used in the development of hardware and software CIs, and their associated documentation. All ITS project members are responsible for implementation of the CM system.

The ITS project is made up of several hardware configuration items (HWCI) and computer software configuration items (CSCI). Configuration control is accomplished by establishing comprehensive baselines at the appropriate time during a program's life cycle. A hierarchy of configuration control boards provides the decision-making and change-tracking mechanisms to ensure that the FDOT and ITS project subcontractors have current configuration status information. The FDOT is responsible for all software development, system hardware, and workstation hardware design. Configuration audits are used to verify that the delivered system performs as specified and matches its documentation.

4.14.1.10 Appendix N – Quality Management Plan

The QMP describes the objectives of how the FDOT or its subcontractors' quality organization plans meet the contract requirements, and satisfy the operational and program goals.

4.14.1.11 Appendix O – Program Performance Management Plan

The program performance management plan describes the methods the FDOT engineers and management, or its subcontractors, use to quantitatively manage the performance of the processes. The quantitative management plan establishes the engineering process goals and product quality goals, as well as the methods used for collecting, analyzing, quantitatively controlling, and reporting performance data in terms of project goals. The quantitative management plan also describes the project management process, using project management measurements for progress and status.

4.14.1.12 Appendix P – Scope of Services

This document is used to specify project work requirements. These requirements are distinctly different from the functional and performance requirements the system must meet, and refer to the requirements the contractor must meet.

4.14.1.13 Appendix Q – Subcontract Management Plan

The subcontract management plan (SMP) describes the process the FDOT uses to select qualified ITS subcontractors and manage them efficiently. The SMP combines the concerns of requirements management, project planning, project tracking, and oversight for basic management control, along with necessary coordination of QA and CM, and applies this control to the subcontractor as appropriate.

4.14.1.14 Appendix R – Concept of Operations

This template is used to create a ConOps document for the project. The ConOps can be a short as a few pages or as big as hundred pages or more depending on the scope and complexity of the project. Often, the ConOps sets the vision for the entire project. It is critical to the success of the ITS project that it reflect the operational vision for the proposed system.

4.14.1.15 Appendix S – Security Engineering Plan

Security engineering is a discipline that focuses on tools, processes, and methods required to design, implement, and test systems that remain dependable in the face of malice, error, or misfortune. This template will help create a plan to address the security issues on a project.

4.14.1.16 Appendix T – Human Factors Engineering Project Plan

The human factor engineering project plan (HFEPP) describes the management plans for the application of HFE design support as related to the man/machine interfaces in FDOT ITS operations.

4.14.1.17 Appendix U – Integrated Logistics Support Plan

The integrated logistics support plan (ILSP) establishes the essential information required to initiate and maintain a through-life integrated logistics support (ILS) program for an FDOT ITS project.

4.14.1.18 Appendix V – Risk Management Plan

This risk management plan (RMP) establishes the process for implementing proactive risk management as part of the overall management of an FDOT ITS project.

4.14.1.19 Appendix W – Reliability and Maintainability Program Plan

The reliability and maintainability program plan (RMPP) describes the necessary tasks, responsibilities, and controls that should be implemented in an FDOT ITS project.

4.14.1.20 Appendix X – System Safety Plan

The system safety plan details the tasks and activities of system safety management and engineering. The plan defines a program to identify, evaluate, and reduce control hazards for a project or system, and its related equipment, facilities, material, services, personnel, and support.

CHAPTER 5 – CONCEPT OF OPERATIONS DOCUMENTATION

5. The Concept of Operations

5.1 Scope

This section provides an overview and guidance for developing a ConOps document for ITS projects. *Appendix R* is a template to be used as a guide in writing the ConOps for a project.

5.1.1 Overview

The FDOT's ConOps guidelines contained in this section utilize the guidelines presented in the *IEEE 1362-1998* standard. This standard, referred to as the *IEEE Guide for Information Technology – System Definition – Concept of Operations (ConOps) Document*, provides the basis for the development of the ConOps document. The structure for the standard's general ConOps document has been modified to support the multifaceted project nature of FDOT ITS projects.

The development of a ConOps document is intended to be part of an initial effort to collect requirements; to develop system concepts and configurations; and to establish how these systems shall operate and interact in the future. This section of *Florida's Statewide SEMP* will provide an overview of the contents of a ConOps document, and a description of the elements and information that should be included in a ConOps document. A ConOps document is often referred to as a "living document" because it reflects a system's evolving requirements. As such, the ConOps document should be reviewed and revised at key milestones during system development. Typically, this includes the inception before system requirements are defined, after the SDR to reflect any changes in the system's operational nature, and during system deployment to support the operations and maintenance manuals used by system operators and managers.

5.1.2 Purpose

This section includes the format and contents to be used when developing or modifying a ConOps document for an ITS deployment or component. In the most general case, an ITS deployment is comprised of hardware, software, people, and manual procedures.

The high-level outline of a ConOps document includes the sections detailed below. Each one of these will be discussed in further detail later in this section. The sections include the:

- Title page
- Document control panel
- Table of contents
- List of figures
- List of tables
- List of acronyms
- Overview and scope
- Referenced documentation
- Current system situation
- Justification and nature of changes
- Concepts for the proposed system
- Operational scenarios
- Summary of impacts
- Analysis of the proposed system
- Notes
- Appendices
- Glossary

Not all projects are concerned with the development of a new system. Some projects consist of a feasibility study and the definition of system requirements. Other projects terminate upon completion of system design or are only concerned with modifications to an existing system. Applicability of the guidelines contained in this section is not limited to projects that develop operational versions of new systems, nor is it limited by project size or scope. Small projects may require less formality than large projects, but all components of this guide should be addressed by every project. The guidelines are meant to be a tool used to assist in the system development process that will allow a thorough understanding of how the system will operate from the users' perspective.

The ConOps approach provides an analysis activity and a document that bridges the gap between the user's needs and visions, and the system developer's technical specifications. In addition, the ConOps provides:

- A means of describing a user's operational needs without becoming bogged down in detailed technical issues that will be addressed during the systems analysis activity

- A mechanism for documenting system characteristics and user operational needs in a manner that can be verified by the user without requiring any technical knowledge beyond that required to perform normal job functions

- A place for users to state their desires, visions, and expectations without requiring the provision of quantified, testable specifications. For example, the users could express their need for a "highly reliable" system and why, without having to produce a testable reliability requirement. In this case, the user's need for high reliability might be stated in quantitative terms by the FDOT prior to issuing a RFP, or the system developer might quantify it during requirements analysis. In any case, it is the responsibility of the FDOT and/or the system developer to quantify user needs.

- A mechanism for users to express thoughts and concerns on possible solution strategies. In some cases, design constraints dictate particular approaches. In other cases, there may be a variety of acceptable solution strategies. The ConOps document allows users to record design constraints, the rationale for those constraints, and an indication of an acceptable solution strategy range.

5.1.3 Responsible Organization

Ideally, representatives of the user community should write the ConOps. In practice, other individuals or organizations may write the ConOps, such as a consultant. In these cases, it is essential that user representatives be involved in providing input, reviewing, revising, and approving the ConOps document. The primary goal for a ConOps is to capture user needs and to express those needs in the user's terminology.

5.1.4 Audience

This guide is intended for users of ITS services, consultants, software developers, system integrators, and other personnel who prepare and update operational requirements for ITS projects and monitor adherence to those requirements.

5.1.5 Evolution of Plans

Developing the initial version of the ConOps document should be one of the first activities completed on a project. As the project evolves, the nature of the work to be done and details of the work will be better understood. The ConOps document should be updated periodically to reflect the evolving situation. Thus, each version of the document should be placed under configuration control.

5.2 Elements of a ConOps Document – A Template to Use

Appendix R is a template for a ConOps document and contains the elements that should be contained in the document as it is developed for a specific project.

This page is intentionally left blank.

CHAPTER 6 –
ENGINEERING SPECIALTY INTEGRATION

6. Engineering Specialty Integration Introduction

Incorporating engineering specialties within the ITS project team increases the expertise available to define the design requirements characteristic of these technical fields. Engineering specialty experts verify that all products within the WBS are designed and fabricated to the specified requirements. Systems engineering ensures that the various engineering specialties perform their tasks efficiently, and that they are integrated into a project from concept design through system installation and support.

Generally, the FDOT engineer manages an ITS deployment project. The FDOT engineer may be assisted by an FDOT project team and GEC staff. In *Florida's Statewide SEMP*, the FDOT person responsible for managing the technical aspects of an ITS project deployment is called the ITS project engineer. The ITS project engineer should be assisted by a test director, who will be responsible for planning and executing the system's IV&V program.

The design, development, and production of a modern ITS product often requires the capabilities and expertise of all engineering and programmatic specialties. The ITS project engineer is often assisted by engineers who are experts in various engineering specialties, such as software, reliability, QM, CM, etc.

Using specialists on the team increases the expertise available to support the incorporation of specialty requirements and characteristics from each discipline into the project. Engineering specialists will verify that all the products within the CWBS are designed and fabricated to support and maintain the requirements throughout their service life.

Engineering specialists included on the ITS team may be provided by the FDOT, other government organizations, contract employees, or contracted consultants. No matter where a specialist's home organization or company is, the specialist will perform project tasks as directed by the ITS project engineer.

6.1 Engineering Specialties

This section identifies and describes the typical engineering specialties that may be applicable to the SEP for an FDOT ITS project. Typical areas of engineering specialties include:

- Reliability and maintainability engineering
- Operability/Human engineering (O/HE)
- Data management engineering
- Software engineering
- Safety engineering
- Test engineering
- Integrated logistics support engineering
- Hardware engineering
- Value engineering (VE)
- Security engineering
- Electromagnetic effects engineering
- Design-to-Cost engineering

6.1.1 Reliability and Maintainability Engineering

The reliability and maintainability (R&M) engineering specialty includes data collection; failure trend analysis; BIT analysis; and the evaluation of specifications and operational R&M parameters.

The reliability engineering effort focuses on the development and application of methods that enhance the safety and reliability of complex technological systems. High priority is given to solutions that consider the task's comprehensive nature for the enhancement of safety and reliability. Reliability activities include such items as:

- Mean time between failure (MTBF) indices
- Technical performance measures and reliability principles
- Models and Monte Carlo simulations
- Pareto distributions for vital problems
- Fault tree analyses (FTA) and design reviews
- Sudden death and simultaneous testing
- Reliability growth models and displays
- Reliability policies, specifications, and audits
- Bathtub curves for failure modes
- Availability, maintainability, and capability
- Weibull, normal, and log-normal probability plots
- Decision trees that merge reliability and costs
- Failure mode effect analysis
- Mechanical components testing for interactions

- Electronic device screening and derating
- Reliability testing strategies and accelerated testing
- Failure recording, analyses, and corrective actions

The maintainability engineering effort is critical in the conception and design phase. It ensures high system availability, where any specific maintainability requirements or goals that must be obtained for a system are identified and the system's maintainability characteristics are quantified. Maintainability characteristics can be represented in terms of a mean time to repair (MTTR), also known as the maximum time to repair, and can be determined for each of the various maintenance levels.

During system development, the maintainability aspect is extremely important and it is vital that developers are aware of their responsibilities in this respect. Maintainability requirements must be expressed as definitively as possible and in quantitative terms of:

- Time (e.g., the time to repair, time between maintenance actions, etc.)
- Rate (e.g., the maintenance hours per operating hours, etc.)
- Complexity (e.g., the number of skill levels, variety of support equipment, etc.)

6.1.2 Operability / Human Engineering

The O/HE specialty is a standardized and formalized approach to the design of human-computer interfaces (HCIs), also called man-machine interfaces (MMIs), which will be optimal, usable, operable, reliable, and fully integrated with the system and software development processes.

When required, HCIs are developed with system software developments to incorporate all system design features and provisions that will enable or enhance the interactions between the system user and the software. These features include the:

- Displays, displayed information, formats, and elements
- Command modes; user-interface languages; and input devices and techniques
- Dialogues; interactions; and transactions between the user and computer
- Online decision aids, procedures, and user documentation
- Provisions for training, prompting, cueing, helping, and tutoring

6.1.3 Data Management Engineering

The data management engineering specialty addresses the overall data management requirements included in a specific project to provide the necessary management and control of the identified operational, management, financial, administrative, or technical data items. Data is essentially anything other than hardware and software, and includes, but is not limited to, drawings, documentation, and source code listings.

The prime functions of data management include such items as:

- Administration of contract deliverables and records
- Data quality and copy control
- Data storage and retrieval systems
- Maintenance and control of supplier-developed and purchaser-furnished information
- Planning, scheduling, and delivery of data

6.1.4 Configuration Management / Quality Control Engineering

The CM/QC engineering specialty addresses the requirements for management of drawings, documentation, and application source codes. This specialty also ensures that all drawings, documentation, and application source codes are complete and prepared according to standard.

6.1.5 Software Engineering

The software engineering specialty defines the activities, objectives, and schedules for system software components during the development life cycle. For large and complex systems, a SDP will be prepared to identify and establish the overall project software management, policies, etc.

6.1.6 Safety Engineering

The safety engineering specialty establishes the organization and defines activities to be used in the identification of possible hazards. This specialty is used to analyze and reduce the risk of hazards occurring relative to the project, including a systems hazard analysis. The major hazards associated with a system that can have an effect on safety are analyzed during a systems hazard analysis. These hazards must be controlled by engineering or eliminated by redesign. The resulting preliminary hazard analysis report contains such items as:

- A brief description of the system, its design, and any subsystems identified
- A list of identified hazards applicable to the system, including descriptions
- A list of identified accidents applicable to the system, including descriptions
- An accident risk classification scheme and probability targets for each accident
- A description of system functions and safety features
- A description of human errors that can create or contribute to accidents
- Conclusions and recommendations

6.1.7 Test Engineering

The test engineering specialty provides a systematic approach for implementing a process to verify that all functional requirements have been complied with. The test engineering specialty establishes a philosophy and strategy for qualifying the system, and includes the identification of any special tests and special test equipment that are needed.

6.1.8 Integrated Logistics Support Engineering

The concepts and techniques associated with the ILS engineering specialty should be applied to all phases of a project's life cycle to ensure that the new or upgraded system will be economically supported through its planned life.

At a minimum, an ILS regimen addresses system hardware and software maintenance; supply support, including resupply and return; spare parts procurement; technical data and documentation; maintenance tools, testing, and support equipment; material transportation and handling; maintenance training; and ILS performance measurements for the life of the program or project. Resources applied to the ILS effort should be scaled to fit the scope and size of the individual project.

The ILS engineering specialty considers all system support factors, evaluating the system's through-life cost (TLC) from any point in the acquisition cycle. Integrated logistics support activities may include:

- Reliability/MTBF predictions
- Reliability block diagrams
- Failure mode effective analysis (FMEA)
- Fault tree analysis
- Sparing calculation and life-cycle costs
- Engineering documentation

6.1.9 Hardware Engineering

The hardware engineering specialty defines the activities, objectives, and schedules for system hardware components during the development life cycle. For large and complex systems, a HDP is prepared to identify and establish the overall project hardware management, policies, etc.

6.1.10 Value Engineering

The VE specialty is an organized effort to obtain optimum value by providing necessary function at the lowest life-cycle cost. This specialty analyzes the functional requirements of systems, equipment, facilities, procedures, and supplies for the purpose of achieving essential functions at the lowest total cost consistent with needed performance, safety, reliability, quality, and maintainability.

The VE specialty can be applied at any point in the system development process but, to obtain maximum effectiveness, VE studies should be undertaken as early as possible, when the impact of decisions on life-cycle costs is the greatest. A VE team usually consists of five to eight persons with diverse backgrounds. The length of time required for a study varies and is dependent on project complexity. Activities to complete during VE studies include those listed in the following sections.

6.1.10.1 Value Engineering Study Activities – Investigation

This activity immediately brings three fundamental VE concepts – function, cost, and worth – to bear on the problem. During the investigation activity, the following basic questions will be asked and answered.

- What is it?
- What does it do or what is its function?
- What must it do or what is its basic function?
- What is it worth?
- What does it cost?

By the end of the investigation activity, the high-cost elements should be identified, functionally analyzed, and assessed for their cost/worth relationships. This activity should also identify the project areas that are candidates for further value study.

6.1.10.2 Value Engineering Study Activities – Speculation

The speculation, or creativity, activity is where the power of the VE technique manifests itself. Creativity is applied to define functional statements that have been selected from cost/worth estimates. The function's generic format is used to speculate on all the possible problem solutions presented in the functional statement.

Brainstorming techniques are applied to develop good alternatives to the current design. This generates a large list of potential problem solutions described by the two-word function. In the next activity, the possible solutions are rapidly pared down to a manageable few through the feasibility analysis.

6.1.10.3 Value Engineering Study Activities – Evaluation

The advantages and disadvantages of each remaining alternative are listed and evaluated. Each advantage and disadvantage is described in general terms, and a weighted matrix analysis is developed for use in selecting the best alternative, based on the relative importance of each to the desirable criteria addressed. This analysis satisfies the VE objective to achieve the best blend of performance, cost, and schedule. Perfection is not the objective of the evaluation activity.

6.1.10.4 Value Engineering Study Activities – Development

Once the best alternative is selected, it is fully developed through sketches, cost estimates, validation of test data, and other technical work to determine if any assumptions made during the study are in fact valid. An implementation plan describing the processes to be followed when implementing any recommendations is prepared.

6.1.10.5 Value Engineering Study Activities – Presentation

The final product of a value study is the formal VE report and the presentation of recommendations. In this activity, the VE findings are presented to the decision makers with justifications for why these ideas should be implemented.

6.1.10.6 Value Engineering Study Activities – Implementation

The decision makers must take appropriate actions to ensure that recommendations are implemented so the anticipated savings are realized.

6.1.10.7 Value Engineering Study Activities – Audits

This activity determines the amount of savings generated by VE, based on the number of recommendations implemented during project development and implementation.

6.1.11 Security Engineering

The security engineering specialty is performed to minimize and contain system vulnerabilities to known or postulated security threats. This is used as definitive guidance in the initial acquisition or modification of new and existing systems, equipment, and facilities to analyze security design and engineering vulnerabilities.

6.1.12 Electromagnetic Effects Engineering

The electromagnetic effects engineering specialty seeks to minimize the deleterious electromagnetic effects in electronic systems by analyzing known system problems to identify possible solutions. This specialty involves the study of systemwide installations to reveal unidentified problems; maintaining a database of known problems and possible solutions; operating a training program; and ensuring that performance specifications address and agree on matters relating to electromagnetic effects. Electromagnetic effects include, but are not limited to, interference with electronic equipment, such as radios; computers; radio frequency (RF) transponders and receivers; message signs; and hazards to personnel or facilities due to lightning, static electricity buildup, or nearby antennas.

The specialty determines and specifies EMC, environmental design, and test requirements. Circuit and electromagnetic field analyses are performed; plans and procedures are prepared; and EMC and magnetic testing is conducted for projects.

6.1.13 Design-to-Cost Engineering

The design-to-cost engineering specialty saves time by stopping the scope growth that leads to schedule extensions. It saves money by limiting overdesign. This specialty emphasizes the use and advantage of functional solutions over prescriptive specifications.

6.2 Engineering Specialty Selection

In most cases, the ITS project engineer needs to be a systems engineer with experience in many areas, including computer-based systems development, operations, and maintenance; transportation facility and roadway design, construction, and operations; and traffic management principles. However, if the ITS project engineer does have expertise in all of these areas, the engineer is usually well-versed in only one area, and has only general knowledge and skills in the others.

Because of this, an experienced ITS project engineer will include engineering specialists on the project team where more than general knowledge about a subject is needed to ensure successful project definition, conceptualization, sizing, design, procurement, implementation, and certification.

This section provides guidelines for selecting the appropriate engineering specialties to include on the project team. *Section 6.3* of this document provides details on a process that can be used as a starting point for identifying the candidate specialties that will be needed for a specific ITS project. A summary of the process is described below.

The four-step process described below is recommended for the ITS project engineer, and possibly the program or project manager, to use when selecting the engineering specialties to include on the ITS project team. The steps include:

- A list of the various project aspects that were defined in the scope documentation should be prepared.
- The list of project aspects should be applied to the matrix in Table 6.1 to identify which engineering specialties may be needed.
- A series of questions for each aspect and engineering specialty should be posted. When answering the questions in this step, the ITS project engineer should record the following.

- A numeric value from 1.00 to 5.00, where:
    - 1.00 = very little
    - 2.00 = somewhat
    - 3.00 = quite a lot
    - 4.00 = a large amount
    - 5.00 = a tremendous amount
- Anything special noted while assigning a value to a question. The series of questions includes the following.
    - How much of the project scope does the aspect comprise?
    - How critical is the aspect to the project in terms of complexity, cost, and schedule?
    - How critical is the aspect in terms of safety and security?
    - Would a specialist in this area contribute significantly to the success of the project?
    - Would not including this specialty significantly limit the success of the project?
    - How much importance does this specialty have regarding the nontechnical aspects of the project, such as the items required by standards or directives?

- The numeric and subjective results should be used to objectively assess how critical this specialty is to the team in achieving success.

It is impractical to arbitrarily assign a "select" or "do not select" value in these *Guidelines*, but the following thresholds are suggested for consideration when deciding which engineering specialties should be included:

- Any single answer with a value of 3.50 or greater
- Any two answers with values of 3.25 or greater
- Any three answers with values of 3.00 or greater
- An accumulated total for all the answers of 14.00 or greater
- A subjective note indicating that sufficient information is not available to objectively answer the questions

6.2.1 Integration of Specialty Engineering Teams

Incorporating engineering specialties within the ITS project team increases the expertise available to define the design requirements characteristic of these technical fields. Engineering specialty experts verify that all products within the CWBS are designed and fabricated according to the requirements, and that the products are supported and maintained throughout their service life. Systems engineering ensures that the various engineering specialties perform their tasks efficiently, and that the specialties are integrated into the project from concept design through system installation and support.

The design, development, and production of the ITS, or engineering, product requires integration across all engineering and programmatic specialties. The following sections address the integration of the engineering specialties in the mainstream system design effort and their integration with other specialties. The system complexity and project schedule dictate a strong interaction with the basic functional and physical design aspects involving systems, software, hardware, and engineering specialties. Many of the specialties have separate project plans, and these plans should be referenced in this section.

6.2.2 Project Integration Meetings

The timely and accurate interfacing of all project specialties is essential to an ITS project's success. Effective and efficient interfacing is accomplished by maintaining detailed plans and schedules that integrate all project activities, and by aggressively implementing the plans. One of the key plan elements is a series of regular formal and informal meetings that include all project specialties. A list of groups that should hold regular meetings applicable to the integration of project specialties is provided below.

- Interface Control Working Group
- Functional Review Group
- Product Assurance Coordination Group
- Integrated Logistics Support Management Team
- Joint Test Group
- Software Engineering Management Team
- Software Design Review Team
- Supportability Assurance Group
- Logistics Support Analysis Review Team
- System Safety Working Group
- Manpower Joint Working Group
- Reliability Working Group

Meeting minutes should be documented and retained as part of the project history archive.

6.3 Engineering Specialty Selection Matrix

Potential ITS project aspects are shown along the left side of Table 6.1 in rows and the specialties are shown across the top of the table in the columns. An X is shown where a project aspect and an engineering specialty that deals with that aspect intersect.

6.3.1 Engineering Specialty Project Integration

Various engineering specialties were described in *Chapter 1*. What separates these engineering specialties from general systems engineering practices is usually a greater depth of knowledge and experience of the subject matter than most engineers possess. Also, most engineering specialties use specialized tools, methods, or calculations that are only appropriate for that specialty. For example, a wind-loading calculation that a hardware engineer will use for a DMS on an overhead gantry will be of no use to a data management engineer who is calculating the computer disk space needed to hold traffic data at a toll plaza.

Although engineering specialties have unique aspects, they are part of the overall SEP. When and how they are used, and to what extent, depends greatly on three things:

- Project size
- Project complexity
- Knowledge and experience of the project engineer

A typical ITS project team will be composed of an ITS project manager, an ITS project engineer, and a test engineer. Figure 6.1 illustrates a typical organization chart with solid lines connecting the functions.

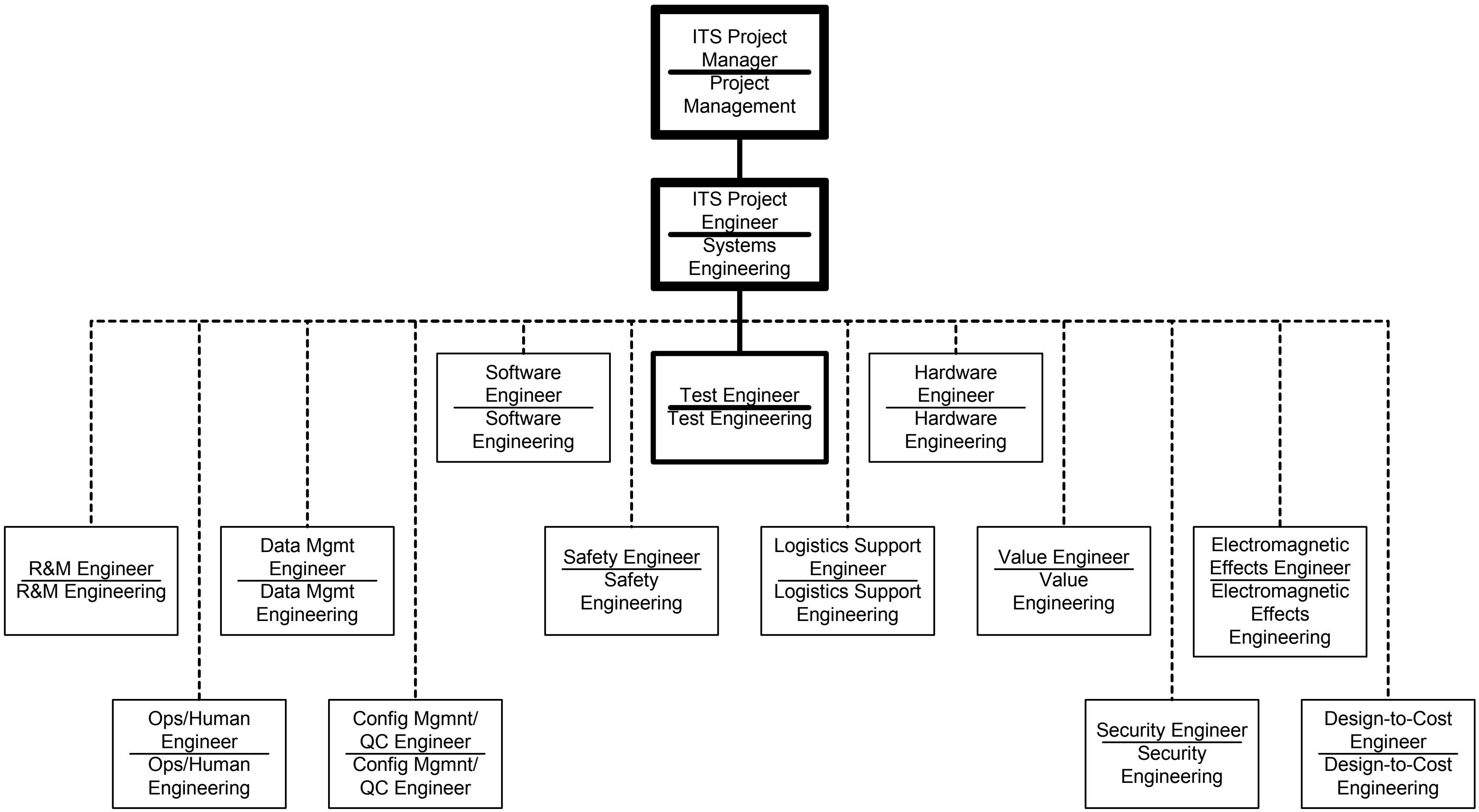
If needed, additional engineering specialties can be added to the team. These are also illustrated in Figure 6.1, with dashed lines connecting the functions. On smaller projects, all engineers will work directly for the ITS project engineer, who will plan, direct, and lead all technical aspects of the project. On larger projects, one or more functional teams, such as software engineering or logistic support teams, may be led by one individual who reports to the ITS project engineer for the group.

An important point that must be understood by all parties is that when integrating engineering specialties in a ITS project team, the engineering specialists perform project tasks for the ITS project engineer, not their home organization or company.

Table 6.1 – Engineering Specialty Selection Matrix

Project Aspects	R&M Engineering	O/HE Engineering	Data Management Engineering	CM / QM Engineering	Software Engineering	Safety Engineering	Test Engineering	ILS Engineering	Hardware Engineering	VE	Security Engineering	Electromagnetic Effects Engineering	Design-to-Cost Engineering
Complex Systems	X	X	X	X	X	X	X	X	X	X	X	X	X
Elevated Work (i.e., Gantries, Towers, Buildings, etc.)						X							
Extensive Project Documentation			X	X									
High Costs (i.e., Materials, Skills, Time, Locations, etc.)								X		X			X
Hardware Fabrication, Assembly, and Installation	X			X			X		X	X			
Hardware Maintenance	X						X	X	X				
In-Traffic Work (i.e., Traffic/Toll Lanes, Road Shoulders, etc.)						X							
Limited Resources (i.e., Budget, Personnel, Materials, etc.)										X			X
Mission-Critical Systems	X			X		X					X		
Proprietary Data (i.e., Financial, Personal, Business, etc.)			X								X		
Severe Environmental Conditions (i.e., Rain, Lightning, Marsh, etc.)	X					X			X	X			
Software Maintenance	X				X		X						
Unattended Sites	X					X		X			X		
User Interface(s)		X		X	X								
Volatile Substances (i.e., Gasoline, Diesel Fuel, Oil, Natural Gas, Explosives, Liquid/Pure Oxygen, Acetylene, etc.)						X						X	

Figure 6.1 – Typical ITS Project Team Organization



6.4 Engineering Specialty Application

As detailed in a previous section, the SEP consists of three phases, with an overarching management and control phase to ensure that the system meets stakeholder expectations throughout its life cycle. Figure 6.2 shows these phases and the typical activities that occur during each phase.

Major events that should occur during the first phase, or the front-end systems engineering activity, include defining the problem, proposing multiple solutions, selecting an optimal solution, defining system requirements based on user needs, and developing specifications that will govern the ITS project's implementation.

The second phase, or the acquisition activity, major events include performing a budget analysis to estimate the system acquisition costs; augmenting the front-end documentation with the budget analysis results; and supporting the documentation development needed to acquire the system or to contract with someone to build it.

The activities that should occur during the third phase, or the back-end systems engineering activity, include the verification and validation of the system solution's compliance with all the requirements defined in the front-end activity.

Ideally, all of the engineering specialties selected for an ITS project will be involved in the front-end phase. It is especially important that each specialist review the project's scope of work, as well as the project definition and performance documentation, to define what the contractual and system functional requirements are in relation to their areas of expertise. This information would then be used for the system specifications documents, which the specialists would later review for completeness and accuracy.

Also during the first phase, applicable engineering specialists would prepare required project documentation (e.g., the test, quality, logistics support, document control, and safety plans). The matrix in Table 6.2 provides a starting point for identifying candidate specialties that may be needed for a specific ITS project phase.

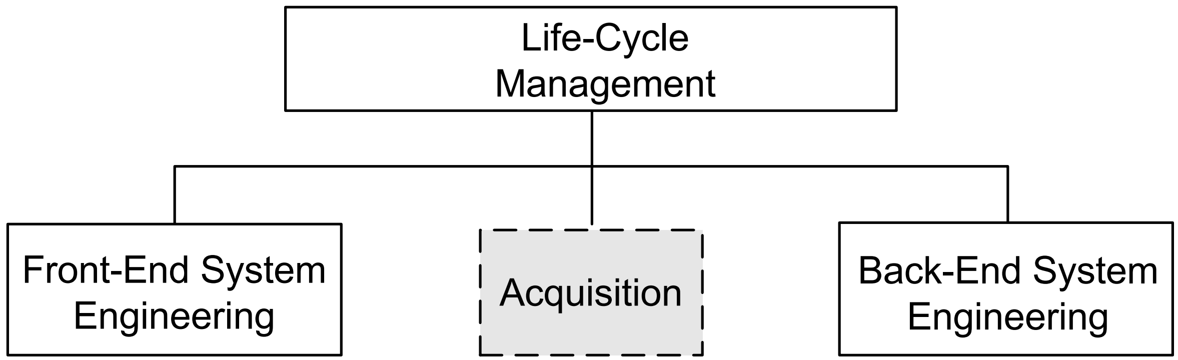
6.4.1 Engineering Specialty Application Matrix

Potential ITS project activities are shown along the left side of Table 6.2 in rows. Engineering specialties are shown across the top of the table in columns. An X is shown where a project activity and an engineering specialty that deals with that activity intersects.

Table 6.2 – Engineering Specialty Application Matrix

Project Aspects	R&M Engineering	O/HE Engineering	Data Management Engineering	CM / QM Engineering	Software Engineering	Safety Engineering	Test Engineering	ILS Engineering	Hardware Engineering	VE	Security Engineering	Electromagnetic Effects Engineering	Design-to-Cost Engineering
Phase I: Front-End Systems Engineering													
Requirements Identification and Analysis	X	X	X	X	X	X	X	X	X	X	X	X	X
Architecture Definition						X							X
Design Synthesis	X	X	X	X	X	X	X	X	X	X	X	X	
Requirements Management			X	X						X			X
System Specifications	X	X	X	X	X	X	X	X	X	X	X	X	X
Phase II: Acquisition													
Scope of Services										X			X
System Specifications			X	X									
Contract Terms and Conditions													
Budget Analysis										X			X
Phase III: Back-End Systems Engineering													
Integration and Test Concept	X	X	X		X	X	X	X	X		X	X	
Integration Testing	X	X	X	X	X	X	X	X	X			X	
Acceptance Testing		X	X	X	X	X	X		X	X			
Configuration Management				X									
Operations and Management Planning	X	X	X		X	X		X					

Figure 6.2 – Systems Engineering Phases



- Requirements Identification and Analysis
- Architecture Definition
- Design Synthesis
- Requirements Management
- System Specifications

- Scope of Services
- System Specification
- Contract Terms and Conditions
- Budget Analysis

- Integration and Test Concept
- Integration Testing
- Acceptance Testing
- Configuration Management
- Quality Control
- Operations and Maintenance Planning
- Training
- Operations Support
- Continuous Product Improvement

CHAPTER 7 –
FLORIDA'S STATEWIDE SEMP SUPPORT PLANS AND PRACTICES

7. *Florida's Statewide SEMP* Support Plans and Practices

7.1 Introduction

Early in the 1990s, it became evident that the life cycles and processes of systems engineering and software engineering were marching along their own separate ways. This was particularly evident in the areas of terminology, design representation, and measurement. The need for a comprehensive, integrated framework to provide better interfaces between systems and software, and to focus on processes that would provide better support to managers of life-cycle stages, was recognized. This framework could then be used to show a common view of systems, their life cycles, and the processes that describe the technical conduct and managerial control required for the framework.[71] The *ISO/IEC 15288-2002(E)*[72] standard provides this framework.

7.1.1 Scope

This chapter defines the various processes used to plan and manage an ITS deployment's entire life cycle. *Chapter 4* of this document describes the specific procedures used to manage the various processes described in this chapter. The processes described herein are adapted from the *ISO/IEC 15288-2002(E)* standard, which is broken into four groups of processes: the agreement, enterprise, project, and technical processes. This chapter will not discuss the agreement process in much detail, since it is mainly an institutional process used to establish intrastate and interstate memorandums of understanding (MOUs) for the use of ITS products.

7.1.2 Purpose

Many business domains, including aerospace, telecommunications, transportation, health care, defense, finance, and information technology, may use the *ISO/IEC 15288* standard. This standard has been adapted to *Florida's Statewide SEMP* and supports *Chapter 4* of this document.

---

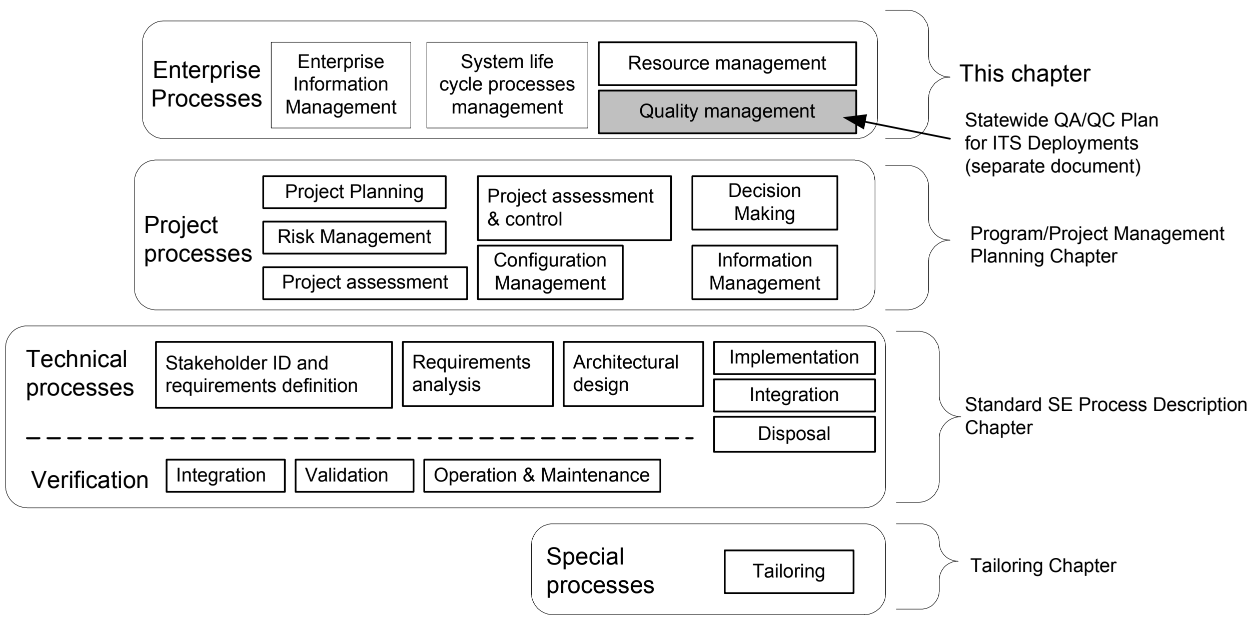
[71] Crowder, Ken, "The End is Here – 15288 to be Published," *INCOSE INSIGHT*, Volume 5, Issue 3 (October 2002): 30-31.

[72] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC), *ISO/IEC 15288-2002 – Standard for Information Technology – System Life Cycle Processes* (October 2002).

7.1.3 Description

The four groups of processes described in the *ISO/IEC 15288* standard include a special tailoring process that is further broken down into a set of 25 detailed processes. Each of the 25 detailed processes is described in terms of its purpose (i.e., the overall goal), outcomes (i.e., the results of the process), and activities (i.e., the decomposition of the process) according to the *ISO/IEC 15288* standard. Figure 7.1 shows the structure of these life cycles and where they are discussed in the *SEMP*. Notice that a complete description of the QM process is provided in a separate document from the *SEMP*. The *ISO/IEC 15288* standard is a high-level standard that describes what should be done throughout a system's complete life cycle. A more detailed description of how to apply the 25 detailed processes is provided in other sections of the *SEMP*, as indicated in Figure 7.1.

Figure 7.1 – Life-Cycle Structure and Relation to *Florida's Statewide SEMP*[73]



[73] Crowder, 31.

7.2 System Life-Cycle Processes

A system's life cycle may be thought of as the different phases that the system goes through from inception to disposal. The entire collection of system phases is often referred to in terms of the human life cycle, such as planning for a system from "birth to death."[74] The degree of formality applied to each process depends on the project and the technology involved. Figure 7.2 depicts the six phases of a system's life cycle.

7.2.1 Concept Phase

In the life cycle's first phase, a need is recognized and a concept is developed for a system that will meet this need. The system is conceived in the concept phase. Stakeholders are identified, the needs are defined, and a ConOps is proposed to satisfy those needs. Often, the ConOps documentation includes early elements of the system architecture, which is expanded on in the life cycle's next phase. (Refer to *Chapter 5*.)

7.2.2 Development Phase

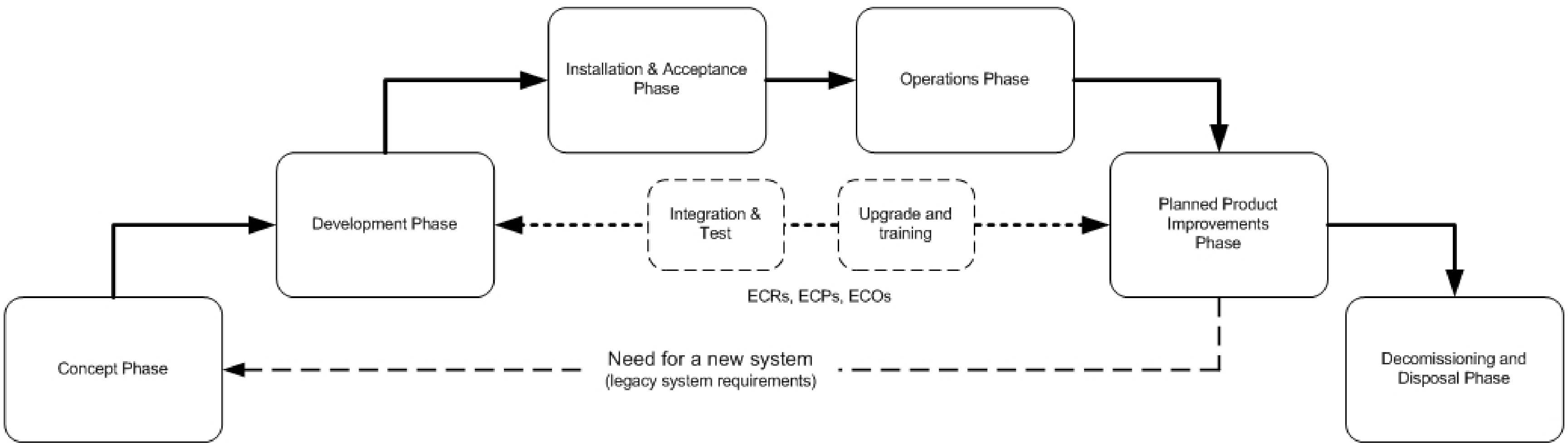
Once the project team has a grasp of the stakeholder needs, a requirements analysis is completed and the system architecture is defined. The development phase is made up of many tasks that are described in detail in *Chapter 4*. This phase is often initiated by the FDOT to define what the system must do. Following the development phase, a procurement process is used to select the contractor that will develop a system that meets the defined requirements.

7.2.3 Installation and Acceptance Phase

At the end of the development phase, the contractor delivers a system that is expected to meet all the requirements. The contractor shall test the system against the requirements and will often have a third party perform an independent test of the system, referred to as an IV&V. After the contractor installs the system, the FDOT will perform or have performed the approved SAT. Training typically takes place during this phase of the system life cycle. Once the system successfully passes all required tests, the Contractor transfers ownership of the system to the FDOT and the next phase of the system life cycle starts.

[74] Another term that is often used is "cradle to grave."

Figure 7.2 – System Life-Cycle Phases



7.2.4 Operations Phase

The system is operated in its specified environment. Any problems that arise are usually fixed by the contractor providing system support or by contractor-trained FDOT personnel who maintain the system. After a period of time, the operational environment changes, subtly at first and more radically later in the operations phase. As the operating environment changes, the system is asked to perform functions that it was never intended to and, eventually, the system should be upgraded.

7.2.5 Planned Product Improvement Phase

When an ITS deployment is planned, consideration should be given to the ease of upgrading the system. For example, the procuring agency may be aware of emerging technology that may be too immature at the time to be considered, but would be desired later as it reaches widespread acceptance. The procurement process should plan for an eventual system upgrade to incorporate that new technology.

In general, systems should be procured that follow standards; have relatively simple and well-defined interfaces; and are modular, allowing the replacement of subsystems without the need to rebuild the entire system. An example would be a software system that uses modular ITS device drivers with a common interface specified by a published ICD. The ITS devices can be added modularly with minimal impact to the system. If the GUI were browser-based, a new ITS device control panel could be added to the GUI once, and then all the users would get the new copy of the interface the next time they logged onto the system.

This is the mature phase of a product's life cycle, and is characterized by improvements initiated through ECRs and ECPs. Approved changes are reviewed by the CMB and implemented through ECOs. (Refer to *Section 3.4*.) The system's improvement will require that the development and integration phases of the system life cycle be repeated for the changes being made. Change will also require documentation updates and, in some cases, training of personnel on how to use the new system features.

7.2.6 Decommissioning and Disposal Phase

This is the end of the system's life cycle. At some point, the system no longer meets stakeholder needs or has become too costly to maintain. In some cases, the manufacturer no longer supports the equipment the system uses. A cost-to-benefit analysis will indicate that it is time to replace the system and the entire system life-cycle process is repeated.

In the concept phase, a decision must be made whether to use any parts of the current system, (i.e., the legacy system), or to scrap the entire system and replace it in whole. Decommissioning involves the process of switching the operations from the legacy system to the new system in a way that allows the enterprise to revert back to the legacy system if the new system doesn't function reliably. At some point after the new system's reliability is proven, the old system is turned off, dismantled, and either put in storage, recycled, or taken to a landfill.

The disposal process must be addressed in the early stages of the system concept phase because it could be an expensive and time-consuming process if not adequately planned for. For example, consider a large piece of machinery that was installed in a building as it was being built. When it came time to replace the machine, a wall had to be removed from the building to get it out. This could have been avoided if the system had been designed using modular machinery that could be disassembled so the parts would fit through a standard office door opening.

7.2.7 Life-Cycle Process Descriptions

This section provides a brief explanation of the following system life-cycle processes:

- Agreement process
- Enterprise process
- Project management process

Project processes will be discussed in greater detail than the first two because most of the *SEMP* is directed toward a process for deploying ITS projects.

7.2.7.1 Agreement Process

According to the *ISO/IEC 15288:2002(E)* standard, the agreement process consists of an acquisition process and a supply process. The agreement process is used to establish agreements with external and internal organizational entities for the acquisition of products or services, or to supply products or services.

7.2.7.1.1 Acquisition Process

This process is used to obtain products or services according to the requirements specified by the organization. *Section 4.2* of this document provides a detailed description of the acquisition process and its relationship with the SEP. This process can be thought of as the creation and issuance of a RFP, request for information (RFI), request for qualifications (RFQ), or ITN.

7.2.7.1.2 Supply Process

This process is described from the viewpoint of the contractor or organization providing the products or services identified in the acquisition process. The supply process is discussed from the FDOT's expectation of how the supplier should respond to the acquisition process in *Chapter 4* of this document.

7.2.7.2 Enterprise Process

The enterprise process describes the activities an organization needs to employ to initiate, support, and control projects. The enterprise process consists of:

- Management of the business environment
- Investment management process
- System life-cycle management process
- Resource management process
- Quality management process

7.2.7.2.1 Management of the Business Environment

The purpose of the enterprise environment's management process is to define and maintain the policies and procedures the FDOT needs to implement *Florida's Statewide SEMP*. It is expected that policies and procedures will be implemented to provide:

- Guidance for long-term, life-cycle management of ITS deployments
- Accountability and authority for system management throughout a product's life cycle
- Policy for planned product improvements throughout their life cycle

An example of an FDOT policy and procedure would be one that addresses QA and QC. For ITS deployment projects, the FDOT's *Statewide QA Plan for ITS Deployments* is to be used to guide QA/QC efforts on ITS deployment projects.

7.2.7.2.2 Investment Management Process

The purpose of the FDOT's investment management process is to initiate and sustain sufficient and suitable project funding to meet the ITS Section's objectives. This process commits the investment of adequate organization funding and resources, and sanctions the authorities needed to establish selected projects. It performs continued qualification of projects to confirm they justify, or can be redirected to justify, continued investment. The FDOT's *Ten-Year ITS CFP* is a good example of the investment management process.

7.2.7.2.3 System Life-Cycle Management

The purpose of the FDOT's system life-cycle management process is to ensure that a system is procured with consideration of the system's life-cycle processes, and that support for those processes is available for the organization's use. Some examples would be the function of the Traffic Engineering Research Laboratory (TERL), which tests and certifies ITS devices for use in Florida. Another example would be the FDOT CSO that supports the acquisition of products and services. A potential support organization could be a central QA/QC office for ITS deployments. This office would ensure that QA/QC policies and procedures were being applied using effective, proven methods and tools. This office could also serve as a resource for the FDOT District ITS engineers as an IV&V for the acceptance of ITS deployments.

7.2.7.2.4 Resource Management Process

This process provides project resources, materials, and services to support organization and project objectives throughout a system's life cycle, including a supply of educated, skilled, and experienced personnel qualified to execute the project. This process assures that there is effective coordination and sharing of resources, information, and technologies throughout the enterprise. Personnel for projects may be drawn from internal FDOT resources or be provided by consultants through the GEC program.

7.2.7.2.5 Quality Management Process

The TEOO ITS Section is expected to apply and administer a QM function that will be identified, with the corresponding QA/QC functions implemented by the Districts throughout the State. The main purpose of the ITS Section's QA/QC function would be to establish quality guidelines according to the *Statewide QA Plan for ITS Deployments* document, and to provide resources and IV&V services to the Districts as requested.

7.2.7.3 Project Management Process

Project management processes are used to establish and manage project plans; to assess actual achievement and progress against the plans; and to control execution of the project through its conclusion. Project management processes can be thought of the as the management procedure for ITS deployments. The FDOT assigns a project administrator who functions as a program or project manager for one or more assigned ITS deployments. Individual project management processes may be invoked at any time in the life cycle and at any level in a hierarchy of projects, as required by project plans or unforeseen events. That is, a project may be created for the management of the spare parts supply for a mature project that is already deployed, or it may be created to manage the competitive procurement of goods and services. Project management processes are applied with a level of rigor and formality that depends on the risk and complexity of the project.

Project management consists of the following processes:

- Project planning
- Project assessment
- Project control
- Decision making
- Risk management
- Configuration management
- Information management

These processes are explained in detail in *Chapter 4* of this document.

7.2.7.4 Technical Processes

Technical processes are used to define the requirements for a system; to design and produce a system that meets the requirements; to support consistent reproduction of the product where necessary; to use the product to provide the required services; to sustain the provision of those services; and to dispose of the product when it is retired from service. Technical processes are described in detail in *Chapter 3*.

This page is intentionally left blank.

CHAPTER 8 –
TAILORING AND SCALABILITY GUIDELINES

8. Tailoring and Scalability Guidelines

8.1 Overview

8.1.1 Purpose

The purpose of the tailoring process is to adapt the systems engineering processes to satisfy particular circumstances or factors that:

- Surround an organization that is employing this International Standard in an agreement
- Influence a project that is required to meet an agreement in which this International Standard is referenced
- Reflect the needs of an organization in order to supply products or services

8.1.2 Description

If this International Standard is tailored, then the organization or project shall implement the activities detailed below according to applicable policies and procedures with respect to the tailoring process, as required. Activities include:

- Identifying and documenting the circumstances that influence tailoring, which include, but are not limited to:
    - Stability of, and variety in, operational environments
    - Risks, commercial or performance, to the concern of interested parties
    - Novelty, size, and complexity
    - Starting date and duration of utilization
    - Integrity issues such as safety, security, privacy, usability, and availability
    - Emerging technology opportunities
    - Development of the profile for the budget and organizational resources available
    - Availability of the services of enabling systems

- In the case of properties critical to the system, considering the life-cycle structures recommended or mandated by standards relevant to the dimension of the criticality

- Obtaining input from all parties affected by the tailoring decisions, which includes, but may not be limited to:

  - System stakeholders
  - Parties interested in an agreement made by the organization
  - Contributing organizational functions

- Make tailoring decisions according to the decision-making process

- Defining a suitable system life-cycle model that permits the system-of-interest to be created and utilized in a manner that conforms to the services needed or the product specified

- Identifying a life-cycle model in terms of stages, their identities, their purposes, and the outcomes they accomplish as a result of the application of the life-cycle processes within each stage, including:

  - The exemplary stages described in this International Standard may be individually selected and used to define the identity, purposes, and outcomes of stages that form part of a selected life-cycle model.

  - Alternatively, life-cycle stages described in this International Standard may be individually selected, identified, and modified, or not applied, as necessary, to achieve changed purpose and outcomes. Document the changes made.

  - Alternatively, define and document any new stage in terms of its identity, purpose and outcomes. Each new stage is assessed to confirm its contribution to a complete and consistent life cycle.

- Select the life-cycle processes that require tailoring in order to satisfy the life-cycle stage outcomes.

  - The life-cycle processes described in this International Standard may be individually selected, identified, and modified, as necessary, to achieve changed purpose and outcomes. Document the changes made; and

  - Alternatively, define and document any new life-cycle process in terms of its identity, purpose, and outcomes. The contribution of each new life-cycle process is evaluated for its contribution to the system.