A103

Introduction to ITS Standards Requirements Development

STANDARDS ITS TRAINING

STUDENT SUPPLEMENT

RITA Intelligent Transportation Systems Joint Program Office

A103: Introduction to ITS Standards Requirements Development

Table of Contents

PURPOSE

This participant outline provides supplemental information for the Professional Capacity Building (PCB) Module A103. In some cases, additional information is included within the supplement. In other cases, references are provided for more in-depth study. Some information may be repeated from the module for context.

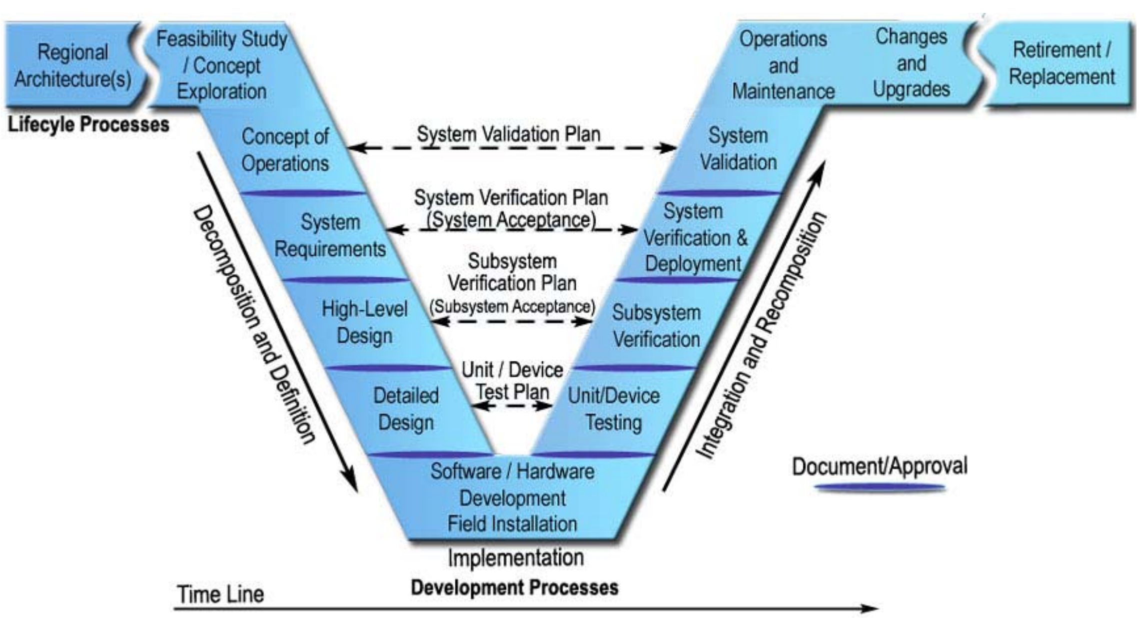
Module A103 introduces what requirements are, where they fit in the project life cycle, how they are used to satisfy user needs and how to determine if they are correct. In A103, participants learn:

1) To define requirements for overall operation to satisfy user needs
2) The concept of a well-formed requirement
3) To define the system and interfaces as a functional architecture
4) To use decomposition of the architecture and requirements as necessary to properly define the system
5) To verify that requirements are complete and correct
6) How requirements development applies to ITS communication standards

The remainder of this supplement is organized based upon these learning objectives and concludes with general references and a glossary.

1 DEFINING REQUIREMENTS FOR OVERALL OPERATION TO SATISFY USER NEEDS

Systems Life-Cycle and the Systems Engineering Process (SEP)



http://ops.fhwa.dot.gov/publications/seitsguide/section3.htm#s3.3
http://ops.fhwa.dot.gov/publications/seitsguide/

Definition of Systems Engineering

- Systems engineering (SE) is an interdisciplinary approach and a means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem: operations, cost, and schedule, performance, training and support, test, manufacturing, and disposal. SE considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. (International Council on Systems Engineering (INCOSE) *Systems Engineering Handbook*)
- An interdisciplinary collaborative approach to derive, evolve, and verify a life-cycle balanced system solution that satisfies customer expectations and meets public acceptability. (Institute of Electrical and Electronics Engineers (IEEE))

Components of a Concept of Operations

- See IEEE Standard 1362-1998.
- See Federal Highway Administration (FHWA) *Systems Engineering Guidebook* V3.
- See ANSI/AIAA G-043-1992.

Definitions of a Requirement

- A statement that identifies a system, product or process' characteristic or constraint, which is unambiguous, clear, unique, consistent, standalone (not grouped), and verifiable, and is deemed necessary for stakeholder acceptability. (INCOSE *Systems Engineering Handbook*)
- A translation of needs into a set of individual quantified or descriptive specifications for the characteristics of an entity in order to enable its realization on examination. (ISO/IEC Guide 25: 1990)

Types of Requirements

- Functional requirements
  - What the system shall do.
  - Example from the Dynamic Message Sign (DMS) Standard

    3.4.2.6 Determine Total Number of Events
    The DMS shall allow a management station to determine the total number of events that the DMS has logged since powerup.

- Performance requirements
  - How well the requirements should perform.
  - Example from the Transportation Sensor System (TSS) Standard

    3.4.2.6 Determine Total Number of Events
    The TSS shall process the Get, GetNext, or Set request in accordance with all of the rules of NTCIP 1103 v02, including updating the value in the database and initiating the transmission of the appropriate response, within 1 second of receiving the last byte of the request.

- Interface requirements
  - Definition of the interfaces.
  - Example from the NEMA TS 2 Standard:

3.3.1.3 Data and Clock Communications Protocol
The transfer of data shall take place over the data and clock links by means of the SDLC (Synchronous Data Link Control) Protocol, as defined by International Business Machines Corporation document GA27-3093-3, dated June 1986.

- Data requirements
  - Definition of the data contained in or interfacing to the system.
  - Example from the Traffic Management Data Dictionary (TMDD) Standard:

3.3.1.4.1.1 Required Error Report Contents
The error report sent from a receiving center to a sending center shall include:

a. Unique identifier of the receiving organization;
b. Unique identifier of the sending organization;
c. Unique error identifier; and
d. Error text.

- Non-Functional requirements
  - Requirements in areas such as reliability, safety, and environmental.
  - Example from the ITS Cabinet Standard:

5.1.26 Operating Ambient Temperature
The TFCS's shall have an operating ambient temperature range from -37 degrees Celsius to +74 degrees Celsius.

- Enabling requirements
  - Requirements that have to do with production, development, testing, training, support, deployment, and disposal.
  - Example from the National Electrical Manufacturers Association (NEMA) TS 2 Standard:

2.1.6 Transients, Power Service
The CA shall maintain all defined functions when the independent test pulse levels specified in 2.1.6.1 and 2.1.6.2 occur on the alternating-current power service.

- Constraints
  - Anything that constrains the development or system that are not already covered by the other areas (e.g. a particular technology, design, tools, and/or standards to be used).
  - Example from the ATC Application Programming Interface (API) Standard:

3.4.2 Programming Language
The API function calls shall be specified using the C programming language as described by "ISO/IEC 9899:1999," commonly referred to as the C99 Standard.

2 WELL-FORMED REQUIREMENTS

Definition of a Well-Formed Requirement

- A statement of system functionality (a capability) that can be validated and that must be met or possessed by a system to solve a customer problem or to achieve a customer objective, and is qualified by measurable conditions and bounded by constraints. (IEEE Standard 1233, 1998 IEEE *Guide for Developing System Requirements*).

WELL-FORMED REQUIREMENT

The Form: [Actor] [Action] [Target] [Constraint] [Localization]

Where:

Actor	Identifies who or what that does the action
Action	Identifies what is to happen
Target	Identifies who or what receives the action
Constraint	Identifies how to measure success or failure of the requirement
Localization	Identifies the circumstances under which the requirement applies

Localization and constraint portions are important but not all requirements will have both

Example: The system [Actor] shall generate [Action] event reports [Target] containing the following information [Constraint] on a scheduled interval [localization]

If a requirement can't be stated in this simple format, you probably need to define the functionality using multiple requirements

<u>Characteristics</u> of Well-Formed Requirements

- Necessary
  - Must be useful and traceable to needs.

- Concise
  - Minimal, understandable, and expressed in a declarative language (i.e., "shall" statements).

- Attainable
  - Realistic to achieve within available resources and time.

- Standalone
  - The requirement is stated completely in one place. Not grouped.

- Consistent
  - Does not contradict itself, nor any other stated requirement.

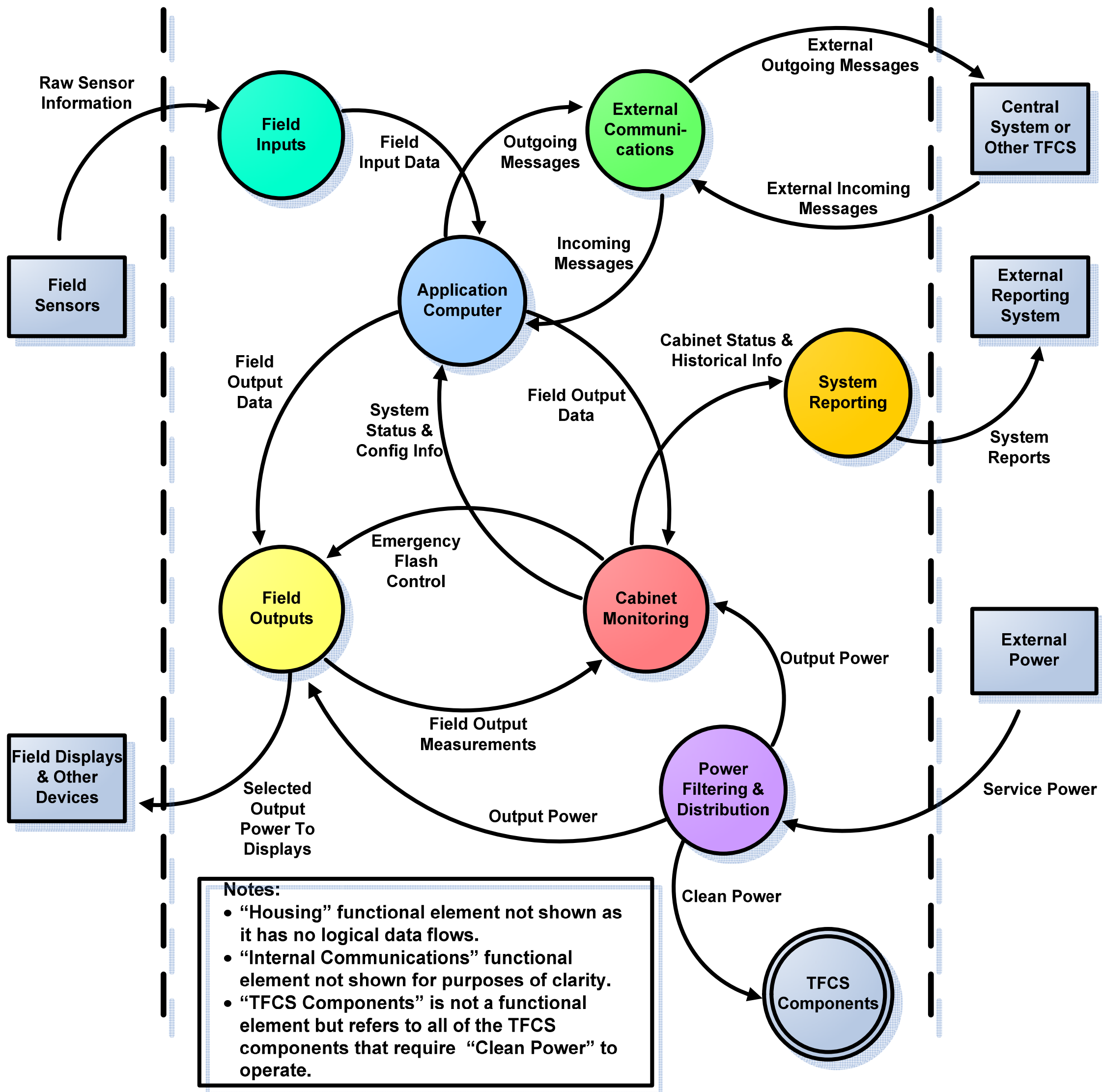
- Unambiguous
  - Susceptible to only one interpretation.

- Verifiable
  - Must be able to determine that the requirement has been met through one of four possible methods: inspection, analysis, demonstration, or test.

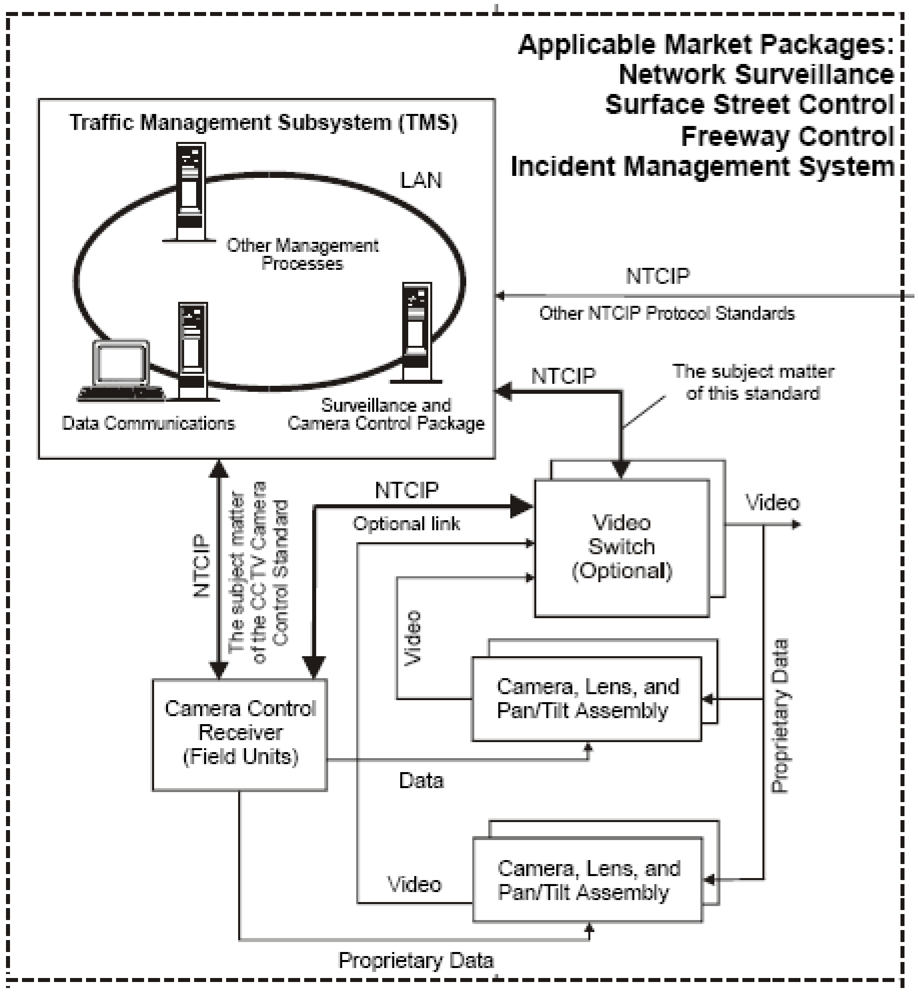
3 DEFINING THE SYSTEM AND INTERFACES AS FUNCTIONAL ARCHITECTURE

Functional Architecture

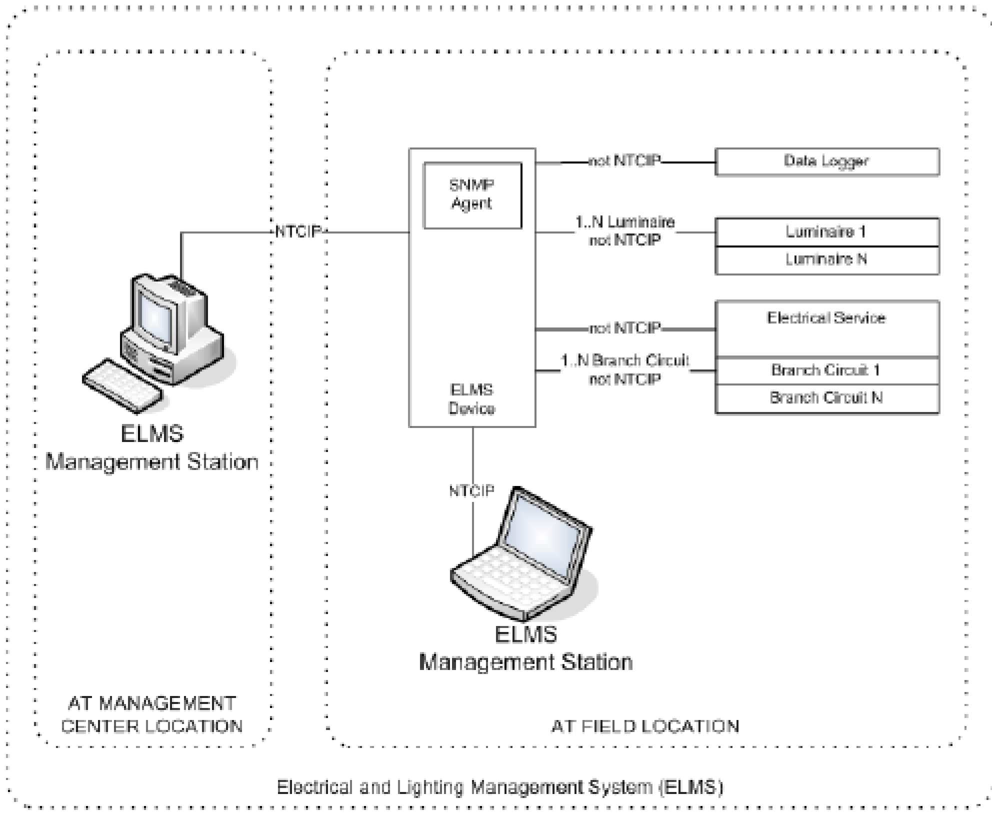
- The parts are sometimes called "functional elements"
- Not a design drawing
- Provides a structure for describing operations in terms of where the operations will be carried out
- Describes what the lines of communication will be
- Many ways to represent an architecture



Example #1 - Architecture Diagram from the ITS Cabinet V2 Standard



Example #2 - Architecture Diagram from the NTCIP 1208 CCTV Standard



Example #3 - Architecture Diagram from the NTCIP 1213 ELMS Standard

4 USING DECOMPOSITION OF THE ARCHITECTURE AND REQUIREMENTS AS NECESSARY TO PROPERLY DEFINE THE SYSTEM

There are many ways to organize requirements. Below are three examples.

Example #1 - By Feature

3.2 System features
3.2.1 System Feature 1
3.2.1.1 Introduction/Purpose of feature
3.2.1.2 Stimulus/Response sequence
3.2.1.3 Associated functional requirements
3.2.1.3.1 Functional requirement 1
. . .
3.2.1.3.n Functional requirement n
3.2.2 System feature 2
. . .
3.2.m System feature m

Example #2 - By User Class

3.2 Functional requirements
3.2.1 User class 1
3.2.1.1 Functional requirement 1.1
. . .
3.2.1.n Functional requirement 1.n
3.2.2 User class 2
. . .
3.2.m User class m
3.2.m.1 Functional requirement m.1
. . .
3.2.m.n Functional requirement m.n

Example #3 - By Mode

3.2 Functional requirements
3.2.1 Mode 1
3.2.1.1 Functional requirement 1.1
. . .
3.2.1.n Functional requirement 1.n
3.2.2 Mode 2
. . .
3.2.m Mode m
3.2.m.1 Functional requirement m.1
. . .
3.2.m.n Functional requirement m.n

5 VERIFYING THAT REQUIREMENTS ARE COMPLETE AND CORRECT

Traceability for Verifying Requirements

- A tool used to <u>help</u> verify completeness and correctness
- Every need must be addressed by at least one requirement
- Every requirement must trace to at least one need
- Any need that is not addressed by at least one requirement means:
  - A requirement was missed, or
  - The user need must be reevaluated
- Every requirement that does not address at least one need means:
  - The requirement must be reevaluated, or
  - A user need was missed
- Every aspect of each user need should be addressed in requirements

There are numerous types of traceability matrices used in ITS Standards, from the simple to the complex. Below are some examples.

Example #1

Requirements Traceability Matrix (RTM) traces requirements to the design elements of the standard. In this case, the design elements include the dialog (sequence of data exchanges) and objects (data elements) that are used to fulfill the requirement.

Requirement ID	Requirement	Dialog ID	Dialog	Object ID	Object
3.4.1.3.8	Execute Pending Configuration				
		4.3.1.3	Execute Pending Configuration Change		
				5.2.1	sensorSystemReset
				5.2.2	sensorSystemStatus
3.4.1.3.9	Abort Pending Configuration				
		4.3.1.4	Abort Pending Configuration		
				5.2.1	sensorSystemReset
				5.2.2	sensorSystemStatus
3.4.1.3.10	Validate Pending Configuration				
		4.3.1.5	Validate a Pending Configuration		
				5.2.1	sensorSystemReset
				5.2.2	sensorSystemStatus

Example #2

Protocol Requirements List (PRL) traces user needs to requirements. It indicates whether the requirement is mandatory, optional, or if there is some conditional conformance to the standard. It then provides a checklist on whether users want to include the requirement in their project. The PRL also provides for other information to be added for further specification or if instructive information is necessary.

User Need Section Number	User Need	FR Section Number	Functional Requirement	Conformance	Support/Project Requirement	Additional Specifications
2.5.2.1	Reset the TSS					
		3.4.1.3.1	Restart the TSS	M	Yes	
		3.4.1.3.2	Reinitialize User Settings	M	Yes	
		3.4.1.3.3	Restore Factory Defaults	M	Yes	
		3.4.1.3.4.	Retune	M	Yes	
		3.4.1.3.8	Execute Pending Configuration	O.1	Yes/No	
		3.4.1.3.9	Abort Pending Configuration	O.1	Yes/No	
		3.4.1.3.10	Validate Pending Configuration	O.1	Yes/No	
2.5.2.2	Initiate Sensor Diagnostics					
		3.4.1.3.6	Short Diagnostics	M	Yes	
		3.4.1.3.7	Full Diagnostics	M	Yes	

6 HOW REQUIREMENTS DEVELOPMENT APPLIES TO ITS COMMUNICATION STANDARDS

Contents of ITS Standards With SEP Content

- General
- Concept of Operations (ConOps)
- Functional Requirements
- Design Details
  - Dialogs and Interface Specifications
  - Object Definitions (MIB)
- Annexes
  - Traceability Matrices
  - Test Procedures (in some standards only)
  - Documentation of Revisions

Contents of ITS Standards *Without* SEP Content

- Overview
- General Information
- Object Definitions (MIB)
- Conformance Groups
- Conformance Statement

PCB Modules on Standards *With* SEP Content

- Dynamic Message Signs
  - A311A Understanding User Needs for DMS Systems Based on NTCIP 1203 Standard
  - A311B Specifying Requirements for DMS Systems Based on NTCIP 1203 Standard
- Environmental Sensor Systems
  - A313A Understanding User Needs for ESS Systems Based on NTCIP 1204 v03 Standard
  - A313B Specifying Requirements for ESS Systems Based on NTCIP 1204 v03 Standard
- Transportation Management Data Dictionary
  - A321A Understanding User Needs for Traffic Management Systems Based on TMDD v03 Standard
  - A321B Specifying Requirements for Traffic Management Systems Based on TMDD v03 Standard

A203 Module on Writing Requirements When ITS Standards Do Not Have SEP – You will:

- Identify different types of requirements
- Understand that requirements development is a process
- Avoid pitfalls when writing requirements
- Write requirements when an ITS communication standard does not have SEP information
- Use traceability matrices as tools for requirements development

ITS Device Communications Standards With SEP Content

- NTCIP 1203 Object Definitions for Dynamic Message Signs (DMS)
- NTCIP 1204 Environmental Sensor Station Interface Standard (ESS)
- NTCIP 1209 Data Element Definitions for Transportation Sensor Systems (TSS)
- NTCIP 1210 Field Management Stations – Part 1: Object Definitions for Signal System Masters (FMS)
- NTCIP 1211 Object Definitions for Signal Control and Prioritization (SCP)
- NTCIP 1213 Object Definitions for Electrical and Lighting Management Systems (ELMS)

ITS Device Communications Standards Without SEP Content

- NTCIP 1201 Global Object (GO) Definitions
- NTCIP 1202 Object Definitions for Actuated Traffic Signal Controller Units (ASC)
- NTCIP 1205 Object Definitions for Closed Circuit Television (CCTV) Camera Control
- NTCIP 1206 Object Definitions for Data Collection and Monitoring (DCM) Devices
- NTCIP 1207 Object Definitions for Ramp Meter Control (RMC) Units
- NTCIP 1208 Object Definitions for Closed Circuit Television (CCTV) Switching
- NTCIP 1212 Object Definitions for Network Camera Operation

REFERENCES

Web sites for ITS Standards

- http://www.standards.its.dot.gov/
- www.fhwa.dot.gov
- www.its.dot.gov/standards/
- www.ntcip.org
- www.ite.org
- www.ieee.org

Web sites for Systems Engineering Development

- www.incose.org
- www.fhwa.org
- www.ieee.org

Guides that Use the Systems Engineering Process

- NTCIP Guide
- TMDD Guide
- IEEE 1512 Guide

Specific Texts, Documents, and Standards

- International Council on Systems Engineering. *Systems Engineering Handbook* Version 3.2. January 2010.
- Alexander, Ian and Ljerka Beus-Dukic. *Discovering Requirements*. Wiley, 2009.
- United States Department of Transportation. *Systems Engineering Guidebook for Intelligent Transportation Systems* Version 3.0. November 2009.
- Berenbach, Brian, Daniel Paulish, Juergen Kazmeier, and Arnold Rudorfer. *Software and Systems Requirements Engineering: In Practice*. McGraw Hill, 2009.
- IEEE 1233-1998 IEEE *Guide for Developing System Requirements Specifications.*
- IEEE 830-1998 *Recommended Practice for Software Requirements Specifications.*
- IEEE Std 1362-1998 IEEE *Guide for Information Technology* – System Definition – Concept of Operations (ConOps) Document.
- ANSI/AIAA G-043-1992 *Guide for the Preparation of Operational Concept Documents.*