

Testing Programs for Transportation Management Systems

A Primer

February 2007



U.S. Department of Transportation
Federal Highway Administration

Notice

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof. This report does not constitute a standard, specification, or regulation.

The United States Government does not endorse products or manufacturers. Trade and manufacturers' names appear in this report only because they are considered essential to the object of the document.

1. Report No. FHWA-HOP-07-089		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Testing Programs for Transportation Management Systems: A Primer				5. Report Date February 2007	
				6. Performing Organization Code	
7. Author(s) Mr. Robert Rausch, P.E., TransCore Ms. Rebecca Barnes, SAIC David Benevelli Mort Serell				8. Performing Organization Report No.	
9. Performing Organization Name and Address TransCore 192 Technology Parkway, Suite 500 Norcross, GA 30092				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No. DTFH61-01-C-00180	
12. Sponsoring Agency Name and Address Operations Office of Transportation Management Federal Highway Administration 400 Seventh Street, SW Washington, DC 20590				13. Type of Report and Period Covered Final Report July 2006 – February 2007	
				14. Sponsoring Agency Code	
15. Supplementary Notes FHWA Task Order Manager: Tom Stout (HOTM)					
16. Abstract Testing Plans for Transportation Management Systems: A Primer is intended to identify for a non-technical audience the key aspects of testing, identify issues for agencies to consider, identify the benefits or value of testing, and profile successful practices for test planning, test procedures, and test execution for the acquisition, operation and maintenance of Traffic Management Systems (TMS) and Intelligent Transportation System (ITS) devices.					
17. Key Word test, testing, acceptance, verification, test plan, installation, transportation management system, intelligent transportation system			18. Distribution Statement No restrictions		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 15	22. Price

This page intentionally left blank.

Table of Contents

Table of Contents	iii
Introduction	1
The Purpose of Testing	1
The Importance of Testing	1
Testing: A Systems Engineering Life-Cycle Process	3
The Basics of Testing	4
Testing Methods	4
Types of Testing	5
Hardware Testing Specifics	7
Software Testing Specifics	8
System-Level Testing Considerations	9
Testing Cost Considerations	11
Additional Resources	11

Introduction

Testing describes a series of processes and procedures developed in the information technology community to verify and validate system performance. While some of the terms used in testing may be unfamiliar to transportation professionals, the core concepts and processes are not technically complex; rather, they represent sound practices in developing and maintaining any system. As you will see in this document, testing is an integral part of deploying a transportation management system (TMS).

The purpose of this primer is to identify for a non-technical audience the key aspects of testing, identify issues for agencies to consider, identify the benefits or value of testing, and profile successful practices for test planning, test procedures, and test execution for the acquisition, operation and maintenance of TMSs and Intelligent Transportation System (ITS) devices.

The Purpose of Testing

There are two fundamental purposes of testing: verifying procurement specifications and managing risk. First, testing is about verifying that what was specified is what was delivered: it verifies that the product (system) meets the functional, performance, design, and implementation requirements identified in the procurement specifications. Therefore, a good testing program requires that there are well-written requirements for both the components and the overall system. Without testable requirements, there is no basis for a test program.

Second, testing is about managing risk for both the acquiring agency and the system's vendor/developer/integrator. The test program that evolves from the overarching systems engineering process, if properly structured and administered, allows managing the programmatic and technical risks to help assure the success of the project. The testing program is used to identify when the work has been "completed" so that the contract can be closed, the vendor paid, and the system shifted by the agency into the warranty and maintenance phase of the project. Incremental testing is often tied to intermediate milestones and allows the agency to start using the system for the benefit of the public.

The Importance of Testing

The guidance provided herein is best utilized **early** in the project development cycle, prior to preparing project plans and acquisition plans. Many important decisions are made early in the system engineering process that affect the testing program. The attention to detail when writing and reviewing the requirements or developing the plans and budgets for testing will see the greatest pay-off (or problems) as the project nears completion. Remember that a good testing program is a tool for both the agency and the integrator/supplier; it typically identifies the end of the "development" phase of the project, establishes the criteria for project acceptance, and establishes the start of the warranty period.

The Benefits of Having Testable Requirements -

Testing is about verifying the requirements; without testable requirements, there is no basis for a test program. Consider the risks and results described in the following two systems:

- ◆ The first system required 2700 new, custom designed, field communication units to allow the central traffic control system to communicate with existing traffic signal controllers. The risk associated with the communication units was very high because once deployed, any subsequent changes would be extremely expensive to deploy in terms of both time and money. As in many TMS contracts, the vendor supplied the acceptance test procedure, the specifications defined the functional and performance requirements that could be tested and the contract terms, and conditions required the test procedure verify all of the functional and performance characteristics. The execution of a rigorous test program in conjunction with well-defined contract terms and conditions for the testing, led to a successful system that continues to provide reliable operation 15 years later.

On the other hand, deploying systems without adequate requirements can lead to disappointment.

- ◆ In the second case, the agency needed to deploy a solution quickly. The requirements were not well defined, and, consequently, the installed system met some but not all “needs” of the acquiring agency. Since the requirements were not well defined, there was no concise way to measure completion. As a result the project was ultimately terminated with dissatisfaction on both the part of the agency and the integrator. Ultimately, the agency decided to replace the system with a new, custom application. However, for this replacement system, the agency adopted a more rigorous system engineering process, performed an analysis of its business practices, and produced a set of testable requirements. The replacement system was more expensive and took longer to construct, but, when completed, it addressed the functional requirements that evolved from the review of the agency’s business practices. Testing assured that the second system met the requirements and resulted in a minimal number of surprises.

In both these cases, testing made a critical difference to the ultimate success of the programs.

Testing: A Systems Engineering Life-Cycle Process

The systems engineering process (SEP) is a methodology and tools for managing a system's life cycle starting with concepts and ending with the system's retirement. It is a highly structured method to facilitate the development, maintenance, refinement, and retirement of dynamic, large-scale systems consisting of both technical components (equipment, information systems, etc.) and human components (users, stakeholders, owners, etc.). Figure 1 is the Systems Engineering "V" Model for ITS that details the various stages that occur within the system's life cycle.

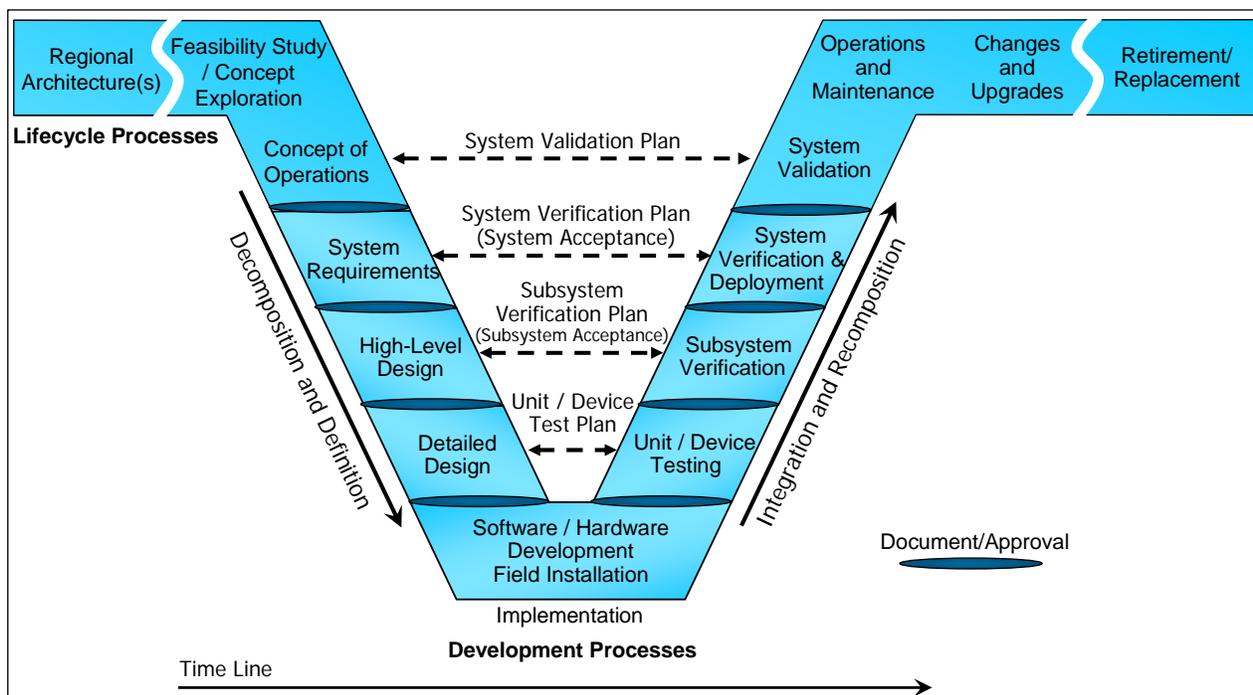


Figure 1. Systems Engineering "V" Model

While testing is shown as **one** stage of the life cycle, it is important to understand that testing is also a **continuous process** within the life cycle. Testing begins with writing the requirements; each requirement must be written in a manner that allows it to be tested. During the design stages, testing will be a consideration as design trade-offs are evaluated for their ability to satisfy the requirements. New requirements may emerge from the designs as choices are made to satisfy the requirements within the project's constraints. Hardware components, software components, subsystems, and systems will be verified during the implementation and testing stages. Final system-level tests will be performed to accept the system and demonstrate the system's readiness for production service. However, testing activities will not end once the system is in operation; testing will continue as the operations and maintenance staff perform corrective, adaptive, and other system maintenance activities.

The Basics of Testing

Testing is the practice of making objective judgments regarding the extent to which the system (device) meets, exceeds or fails to meet stated objectives. This section provides a basic tutorial of testing methods, test types and test execution. It is not of sufficient depth or breath to make you an expert, but it will introduce you to the concepts and terminology that are likely to be used by the system vendors throughout the test program.

Testing Methods

The complexity and ultimate cost of the system test program is directly related to the test method(s) specified for verification of each requirement. As the test method becomes more rigorous, performing that testing becomes more time consuming, requires more resources and expertise, and is generally more expensive. However, the risk of accepting a requirement under a less rigorous testing method may be undesirable and may ultimately prove to be more expensive than a more rigorous test method. There are five basic verification methods, as outlined below.

- ◆ **Inspection** - Inspection is the verification by physical and visual examinations of the item, reviewing descriptive documentation, and comparing the appropriate characteristics with all the referenced standards to determine compliance with the requirements. Examples include measuring cabinets sizes, matching paint color samples, observing printed circuit boards to inspect component mounting and construction techniques.
- ◆ **Certificate of Compliance** - A Certificate of Compliance is a means of verifying compliance for items that are *standard products*. Signed certificates from vendors state that the purchased items meet procurement specifications, standards, and other requirements as defined in the purchase order. Records of tests performed to verify specifications are retained by the vendor as evidence that the requirements were met and are made available by the vendor for purchaser review.
- ◆ **Analysis** - Analysis is the verification by evaluation or simulation using mathematical representations, charts, graphs, circuit diagrams, calculation, or data reduction. This includes analysis of algorithms independent of computer implementation, analytical conclusions drawn from test data, and extension of test-produced data to untested conditions. An example of this type of testing would include internal temperature gradients for a dynamic message sign. It is unlikely that the whole sign would be subjected to testing within a chamber, so a smaller sample is tested and the results can be extrapolated to the final product.
- ◆ **Demonstration** - Demonstration is the functional verification that a specification requirement is met by observing the qualitative results of an operation or exercise performed under specific condition. This includes content and accuracy of displays, comparison of system outputs with independently derived test cases, and system recovery from induced failure conditions.

- ◆ **Test (Formal)** - Formal testing is the verification that a specification requirement has been met by measuring, recording, or evaluating qualitative and quantitative data obtained during controlled exercises under all appropriate conditions using real and/or simulated stimulus. This includes verification of system performance, system functionality, and correct data distribution.

It is important to carefully consider the tradeoffs and consequences attendant to the specification of the test methods since they flow down with the requirements to the hardware and software specifications and finally to the procurement specification. From a verification standpoint, **inspection** is the least rigorous method, followed by **certificate of compliance**, **analysis**, **demonstration**, and then **test** (formal) as the most rigorous method. A vendor's certificate of compliance may be evidence of a very rigorous development and test program, but that testing is typically not defined, approved, or witnessed by the system's acquiring agency; therefore, there is some risk in accepting that certification. That risk is often outweighed, however, by the costs of performing equivalent testing by the acquiring agency. Remember that the vendor's test program costs are embedded in the final component pricing.

Types of Testing

Several levels of testing are necessary to verify the compliance of all of the different system requirements. Multi-level testing allows verification of compliance to requirements at the lowest level, building up to the next higher level, and finally full system compliance with minimal re-testing of lower level requirements once the higher level testing is performed. A typical system test program has five levels of verification tests, as outlined below.

- ◆ **Unit Testing -**

Hardware Unit Testing - The hardware component level is the lowest level of hardware testing. Hardware unit tests are typically conducted at the manufacturing plant by the manufacturer. At this level, the hardware design is verified to be consistent with the hardware detailed design document.

Software Unit Testing - The computer software component level is the lowest level of software testing. Stand-alone software unit tests are conducted by the software developer following design walk-throughs and code inspections. At this level, the software design is verified to be consistent with the software detailed design document.

- ◆ **Installation Testing -**

Delivery Testing - Delivery testing is conducted to verify that all of the equipment has been received without damage and is in accordance with the approved design. This testing is performed by the agency or its installation or integration contractor, who must receive and store the devices and integrate them with other ITS devices before installation.

Site Installation Testing – Site installation testing is performed at the installation site subsequent to receiving inspections and functional testing of the delivered

components. Here, checklists are used to assure that any site preparation and modifications, including construction, enclosures, utilities, and supporting resources have been completed and are available. Specific emphasis on test coordination and scheduling, particularly for the installation of communications infrastructure and roadside components, is essential to achieving successful installation testing and minimizing the associated cost and should be detailed in procurement specifications.

◆ **Integration Testing -**

Hardware Integration Testing - Integrated hardware testing is performed on the hardware components that are integrated into the deliverable hardware configuration items. This testing can be performed at the manufacturing facility (factory acceptance tests) or at the installation site, as dictated by the environmental requirements and test conditions stated in the test procedures.

System Software Build Integration Testing - Software build integration testing is performed on the software components that are combined and integrated into the deliverable computer software configuration items. A software build consisting of multiple computer software configuration items is ideally tested in a separate development environment.

Hardware Software Integration Testing - Once the hardware and computer software configuration items have been integrated into functional chains and subsystems, hardware/software integration testing is performed to exercise and test the hardware/software interfaces and verify the operational functionality in accordance with the specification requirements. Integration testing is performed according to the integration test procedures developed for a specific software release. Testing is typically executed on the operational (production) system unless the development environment is sufficiently robust to support the required interface testing.

◆ **Regression Testing –**

Regression testing is required whenever hardware and/or software are modified, and when new components and/or subsystems are incorporated into the system. Regression tests are necessary to assure that the modifications and added components comply with the procurement functional and performance specification requirements and that they perform as required within the integrated system without degrading the existing capability. Regression tests are usually a subset of a series of existing tests that have been performed, but may require a complete retest. Sufficient testing should be performed to re-test the affected functionality and system interface compatibility with the modified or new component. The goal of regression testing is to validate in the most economical manner that a modification or addition does not adversely impact the remainder of the system. It is important that the tests regress to a level of tests and associated test procedures that include the entire hardware and software interface. In most cases, this will require some unit level interface and integration testing in addition to functional and performance testing at the subsystem and total system level.

◆ **Acceptance Testing –**

Subsystem Acceptance Testing - Acceptance testing at the subsystem and system level is conducted by the acquiring agency prior to contractual acceptance of that element from the developer, vendor, or contractor. Acceptance testing of the installed software release is performed on the operational system to verify that the requirements for this release have been met in accordance to the system test procedures. Two activities must be completed in order to commence subsystem acceptance testing. All lower level testing, i.e., unit, installation, and integration testing, should be complete. Additionally, any problems identified at these levels should be corrected and re-tested. Alternatively, the agency may wish to make changes to the requirements and procurement specifications to accept the performance or functionality as built and delivered.

System Acceptance Testing - Acceptance testing at the system level should include an end-to-end or operational readiness test of sufficient duration to verify all operational aspects and functionality under actual operating conditions. While it may not be possible to test all aspects of the required system level functionality in a reasonable period of time, the system test plan should specify which of these requirements must be tested and which should be optionally verified given that those operational circumstances occur during the test period. The acquiring and operating agencies must be ready to accept full operational and maintenance responsibilities (even if some aspects of the operations and maintenance are sub-contracted to others). This includes having a trained management, operations and maintenance staff in place prior to the start of the system acceptance testing.

After components are tested and accepted at a lower level, they are combined and integrated with other items at the next higher level, where interface compatibility, and the added performance and operational functionality at that level, are verified. At the highest level, system integration and verification testing is conducted on the fully integrated system to verify compliance with those requirements that could not be tested at lower levels and to demonstrate the overall operational readiness of the system.

Hardware Testing Specifics

The hardware test program is intended to cover the device testing from prototype to final deployment. The degree of hardware testing required for a TMS will depend on the maturity and track record or installation history of the device (product), the number of devices purchased, the cost of the testing, and the risk of system failure caused by problems with the device. For classification purposes, the maturity of the device will be categorized based on its history, which can vary from *standard* devices (typically standard product), to *modified* devices, and to *new or custom* devices developed for a specific deployment. In general, the hardware test program can be broken into six phases as described below.

- ◆ **Prototype testing** – Prototype testing is generally required for “new” and custom product development but may also apply to modified product depending on the na-

ture and complexity of the modifications. This tests the electrical, electronic, and operational conformance during the early stages of product design.

- ◆ **Design Approval Testing (DAT)** – DAT is generally required for final pre-production product testing and occurs after the prototype testing. The DAT should fully demonstrate that the ITS device conforms to all of the requirements of the specifications.
- ◆ **Factory Acceptance Testing (FAT)** – FAT is typically the final phase of vendor inspection and testing that is performed prior to shipment to the installation site. The FAT should demonstrate conformance to the specifications in terms of functionality, serviceability, performance and construction (including materials).
- ◆ **Site Testing** – Site testing includes pre-installation testing, initial site acceptance testing and site integration testing. This tests for damage that may have occurred during shipment, demonstrates that the device has been properly installed and that all mechanical and electrical interfaces comply with requirements and other installed equipment at the location, and verifies the device has been integrated with the overall central system.
- ◆ **Burn-In and Observation Period Testing** – A burn-in is normally a 30 to 60 day period that a new device is operated and monitored for proper operation. If it fails during this period, repairs or replacements are made and the test resumes. The clock may start over at day one, or it may resume at the day count the device failed. An observation period test normally begins after successful completion of the final (acceptance) test and is similar to the burn-in test except it applies to the entire system.
- ◆ **Final Acceptance Testing** – Final acceptance testing is the verification that all of the purchased units are functioning according to the procurement specifications after an extended period of operation. The procurement specifications should describe the time frames and requirements for final acceptance. In general, final acceptance requires that all devices be fully operational and that all deliverables (e.g., documentation, training) have been completed.

Testing is about controlling risk for both the vendor and the agency. For most TMS projects, standard ITS devices will be specified, so the required hardware test program will be made up of a subset of the above listed tests. It should be noted that there are many variations of burn-in period, final (acceptance) testing, and observation period and the agency should think the process through thoroughly and understand the cost/risk implications.

Software Testing Specifics

The software test program is intended to cover the software testing from design reviews through hardware/software integration testing. All software proposed for use in your TMS should be subjected to testing before it is accepted and used to support opera-

tions. The extent and thoroughness of that testing should be based on the maturity of that software and the risk you are assuming in including it in your operations.

For most agencies, the bulk of software testing, at various levels of completeness, will be done at the software supplier's facility and the agency will not participate. It is suggested that the procurement specifications should contain provisions that give the agency some confidence that a good suite of tests are actually performed, witnessed and properly documented. This may require the software supplier to provide a description of their software quality control process and how the performance of the tests for the agency's project will be documented. The agency should review the supplier's documented quality control process and sample test reports, and be comfortable with the risk associated with accepting them. Where possible, the agency should plan to send personnel to the developer's facility periodically to observe the progress and test the current "build" to ensure that the translation of requirements to code meets the intended operation.

In general, the software test program can be broken into three phases as described below.

- ◆ **Design Reviews** – There are two major design reviews: (1) the preliminary design review conducted after completion and submission of the high-level design documents and (2) the detailed design (or critical) review conducted after submission of the detailed design documents.
- ◆ **Development Testing** – For software, development testing includes prototype testing, unit testing, and software build integration testing. This testing is normally conducted at the software developer's facility.
- ◆ **Site Testing** – Site testing includes hardware/software integration testing, subsystem testing, and system testing. Some integration testing can be conducted in a development environment that has been augmented to include representative system hardware elements (an integration facility) but must be completed at the final installation site (i.e., the transportation management center) with communications connectivity to the field devices.

As with the hardware test program, the software test program is also dependent on the procurement specifications. The procurement specifications must establish the requirements for the contract deliverables and the testing program, specify the consequence of test failure, and identify the schedule and cost impacts to the project.

System-Level Testing Considerations

System-level testing is typically conducted to verify both completed subsystems and the system as a whole from the software developer, vendor(s), and systems integration contractor under the terms of the contract. Before attempting system-level testing, all unit, installation, and hardware/software integration testing should be complete. Any problems identified at these levels should have been corrected and re-tested. It is also

possible that the agency has decided to accept certain changes; under these circumstances, it is important that the system requirements be changed and documented and that the revised requirements serve as the basis for the final systems testing.

If the TMS is being incrementally deployed, then the system-level test planning and procedures must be developed such that each increment can be separately tested and accepted. Under these circumstances, significant regression testing may also be required to ensure that the incremental functionality or geographical extension does not compromise the operation of the system. In general, the system-level test program can be broken into two phases as described below.

- ◆ **Subsystem Testing** - Subsystem verification testing is performed as a prelude to system testing. It is performed in the operational environment using installed system hardware and software. Testing at the subsystem level should be performed: a) when different developers, vendors, or contractors have been responsible for delivering stand-alone subsystems, b) when the complete functionality of a subsystem could not be tested at a lower level because it had not been fully integrated with the necessary communication infrastructure, or c) when it was previously impossible to connect to the field devices for the testing phase.

Testing at this level has distinct benefits over delaying that testing to the higher level system testing:

- The test procedures and test personnel can concentrate on a limited set of system requirements and functionality.
 - Problems encountered during the test can be resolved independent of other testing.
 - Testing can be completed in a shorter time span and with fewer resources and disruption to other operations.
 - Acceptance can be incrementally achieved and vendors paid for completed work.
- ◆ **Systems Testing** - System verification testing is the highest test level; it is also usually the one with the fewest requirements remaining to be verified. Only those requirements relating to subsystem interactions, quantity of field devices, external interfaces, and system performance should remain to be formally verified. System acceptance testing is performed after all lower level testing has been successfully completed. It is performed in the operational environment using all available and previously installed and tested system hardware and software. The system acceptance test should include an end-to-end or operational readiness test of sufficient duration to verify all operational aspects and functionality under actual operating conditions. Formal acceptance at the subsystem or system level may trigger the start of equipment warranty periods, software licensing agreements, operations and maintenance agreements, etc. The procurement documents should clearly specify which of these are applicable, when they become effective and need to be renewed, and what the payment schedules and provisions are.

It is very important that the procurement document clearly and unambiguously describe what constitutes final acceptance. Procurement documents frequently require the conduct of an acceptance test followed by an observation period. The contractor's expectation is that title to the system transfers at the completion of the test, but the agency intended the transfer to occur after completion of the observation period. The transfer can occur any way the agency wants, but it should be uniformly understood by all parties.

Testing Cost Considerations

The test location, test complexity, number and types of tests, and the test resources required (including test support personnel, system components involved, and test equipment) impact testing costs. Testing is expensive and represents a significant portion of the overall TMS acquisition budget, but this should not dissuade you from conducting a thorough and complete test program that verifies that each of your requirements has been met. You ultimately control testing costs by the number and specificity of your requirements. A small number of requirements with minimum detail will be less costly to verify than a large number of highly detailed requirements. While both sets of requirements may result in similar systems, the smaller, less complex set takes less time and resources to verify. There are more opportunities for a quality product at the lowest cost if the procurement documents specify only the functionality and performance characteristics to be provided, leaving the design and implementation decisions up to the contractor. However, this approach should be balanced with the agency's expectations since there may be various means by which the requirement *could* be satisfied by a vendor.

Additional Resources

The handbook entitled *Testing Programs for Transportation Management Systems* is intended to provide direction, guidance, and recommended practices for test planning, test procedures, and test execution for the acquisition, operation, and maintenance of transportation management systems and ITS devices. The handbook expands on the information presented here, by detailing the various aspects and components of testing.

The documents listed below and additional training materials are available on the Federal Highway Administration's website at <http://ops.fhwa.dot.gov/publications/publications.htm/>.

- ◆ Federal Highway Administration, *Testing Programs for Transportation Management Systems – A Technical Handbook*, (Washington, DC: 2007).
- ◆ Federal Highway Administration, *Testing Programs for Transportation Management Systems – Practical Considerations*, (Washington, DC: 2007).

Testing Programs for Transportation Management Systems: A Primer



Federal Highway Administration
U.S. Department of Transportation
400 7th Street, S.W. (HOP)
Washington, DC 20590
Toll-Free "Help Line" 866-367-7487
www.ops.fhwa.dot.gov

Publication No.: FHWA-OP-07-089
EDL Document No.: 14383