

A 3 2 1 b

**Specifying Requirements
For Traffic Management
Systems Based
on TMDD v3 Standard**



STUDENT SUPPLEMENT

RITA Intelligent Transportation Systems
Joint Program Office

A321b
**Specifying Requirements for Traffic
Management Systems Based on TMDD v3.0 Standard**

Table of Contents

Introduction	2
Example of Areas of Operational Needs Covered by TMDD	3
System Interface.....	5
Source of User Needs, Requirements, and Data Concepts.....	6
Preparing System-Interface Specifications	17
Sample Requirements Traceability Matrix (RTM)	21
Application-Level Protocols.....	22
What if a Requirement is not found in the TMDD Standard?	25
Extensions	26
References.....	27

Introduction

This supplement provides additional information on the materials covered in Module **A321b Specifying Requirements for Traffic Management Data Dictionary (TMDD) v3.0 Standard**. This document aids further in amplifying the discussion issues covered in the course. A321a, the first course on the TMDD v3.0 standard focused on the TMDD user needs. This course focuses on requirements related to the TMDD standard.

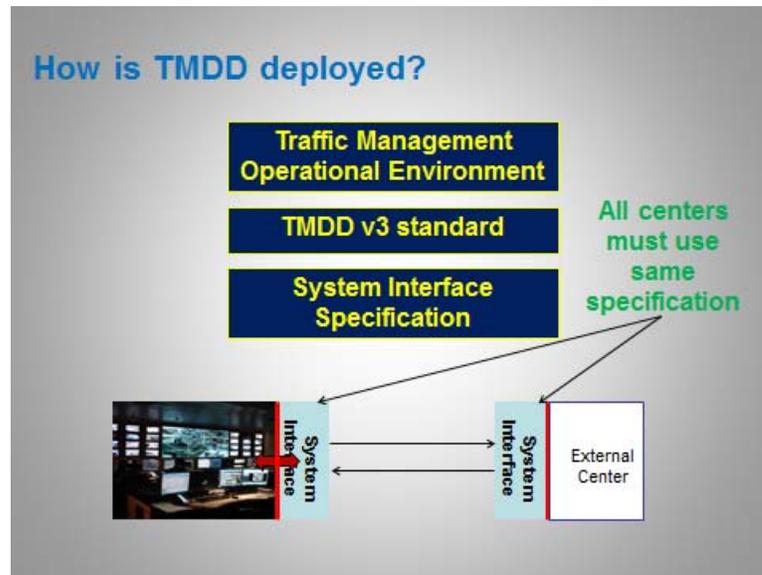


Figure 1: Role of TMDD in System Interface Specification

The TMDD is an information level standard. This dictionary serves the traffic management area of intelligent transportation systems (ITS). It supplies building blocks or design materials from which a system interface is designed. As shown in Figure 1, the TMDD standard supplies standardized definitions of user needs and their requirements and associated design data concepts. These standardized data concepts are dialogs messages, data frames, and data elements. The standard has complied definitions. Specification writers must only modify their needs for a local project. The TMDD standard enables a system-interface design that meets interoperability needs of a traffic management environment. The system interface, a software product, serves the needs of traffic management in the center to center (C2C) context. The reader is directed to the reference section for C2C case studies relevant to the course discussion topics: *user needs, requirements, and specification for C2C system interface*. (See final report: <http://www.aztech.org/docs/c2c/az-c2c-conops.pdf>).

Example of Areas of Operational Needs Covered by TMDD

Centers typically have operational needs in the four key areas to exchange information. These four areas are selected out of eight covered by the standard, as shown in Figure 2, because they describe how interconnected center-to-center (C2C) operational context supports traffic management functions.

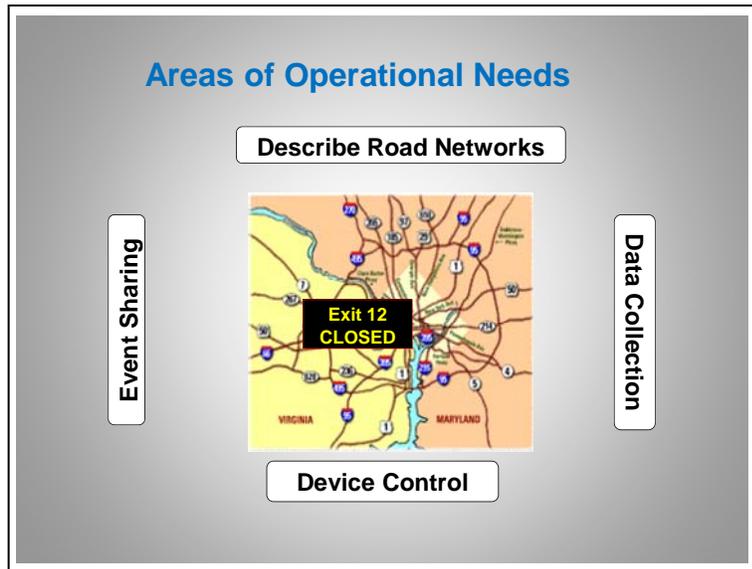


Figure 2: Key Areas of Operational Needs Covered by TMDD
(Other four areas related to network administration)

- **Facilitate Remote Command and Control of Field Devices:** For example, as shown in Figure 1, a neighboring EC (through prior arrangements) may be able to remotely control certain traffic devices that belong to another jurisdiction during after-hours or when a TMC may be closed or at other times may “command” to take some action. Although such occurrences may not be common, during a major emergency such as a natural disaster in the region, such a need may arise.
- **Share/Exchange Event Information:** Information exchange is key to improved operational coordination and collaboration in the regional context. For example, within a C2C operational environment, the TMC and public safety external center often have information that is valuable to each other’s operations. These centers often desire system-to-system communication to exchange information that will help them coordinate responses to transportation conditions that have regional impacts. The types of information may include incident event information or traffic congestion levels during a planned event.

- **Provide Coordinated Response to Transportation Conditions:** For example, a local traffic incident may have an impact on a region that necessitates a change in current signal timing on an arterial passing through multiple jurisdictions. In such a case, all TMCs in the region will attempt to coordinate a pre-planned or new response pattern.
- **Share Roadway Network Data:** For example, participating centers can make available to each other information on their roadway network, such as nodes, links, and routes.

The above discussion can also be summarized in a C2C context in the following manner:

- TMDD identifies the operational needs and provides necessary and required design materials to develop a system interface that has the capability to:
 - Manage assets and other entities
 - Manage information
 - Monitor status
 - Control devices

System Interface

The IEEE Std. 610, Glossary of Software Engineering Terminology defines a system interface as “a shared boundary across which information is passed.”

As shown in Figure 3 by the RED vertical bar, a system interface creates a boundary with a local system at the TMC. This boundary allows information across, with the local traffic management system, and to the outside world, which could be a TMC or emergency management.

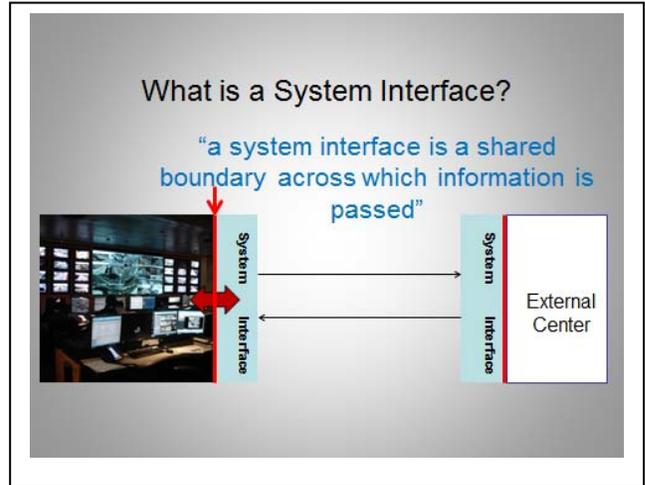


Figure 3: System Interface

To achieve the above objective, users have to first write a specification to procure and develop a system interface. A system interface is NOT a commercial off-the-shelf (COTS) software product. It is a customized software entity; not one size fits all.

This is illustrated in Figure 4 in terms of how messages are sent and responses are received through a system interface. These messages are created using TMDD-supplied standardized data concepts definitions. Recognizing that project needs are different both in scale and user needs from place to place, system interfaces must be developed to meet local needs. As a result of a variance in local needs, the development of the system interface (complexities) varies from place to place.

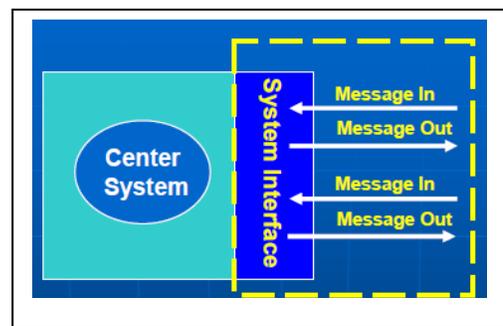


Figure 4: Messages thru an Interface

For example, a small city TMC may just have an operational need for only a traffic controller’s related function exchanges with a neighboring jurisdiction to coordinate signal timing strategy at boundaries, and messages will be limited to such tasks only, while a statewide TMC may need to perform a range of ATMS functions that will require support for a large number of messages .

Source of User Needs, Requirements, and Data Concepts

As shown in Figure 5, Volume I and Volume II together make up the TMDD v3.0 standard. The standard documentation organization follows this sequence—*User Needs-Requirements-Data Concept*—and guides the reader through specification preparation for the system interface consistent with the systems engineering approach. In this layout, Volume I deals with addressing a known C2C problem, and Volume II provides for solution content that a separate design phase of the project will use.

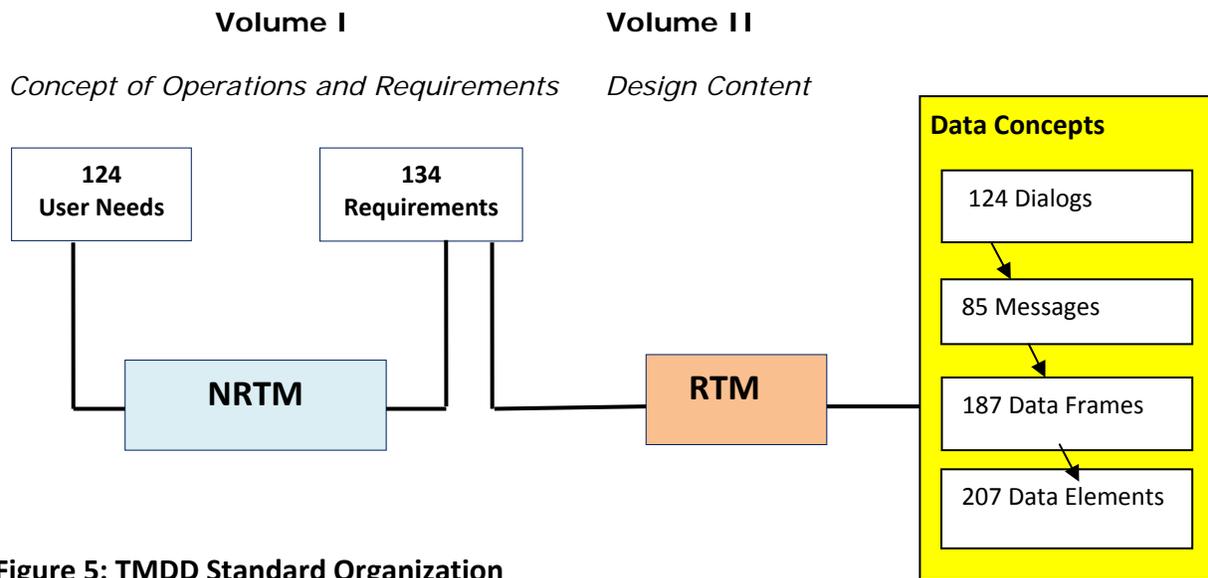


Figure 5: TMDD Standard Organization

The user should view this diagram with the NRTM and RTM in mind as tools to prepare a project specification and “building materials” available in order to design system interfaces. Notice the large inventory of definitions compiled by the standard available for any situation a user wishes to address. (This conceptual representation is indicative of a “shopping cart” analogy; take what you need.)

As shown in Figure 5, NRTM is the only way to select standard-supplied user needs and tailoring requirements that satisfy the selected user needs. (A requirement is a condition or capability, “solidly” expressed in “shall” language. Requirements link us to the next step—design content.) Similarly, only RTM necessary specific-design concepts (solutions) that fulfill the requirements are selected. As shown in the figure, there are no direct links between user needs and requirements. Thus, the TMDD standard has ensured traceability in both forward and backward direction of the development process, in addition to providing for standardized definitions of user needs, requirements, and design concepts.

Data Concepts

Data is a representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by human or by automatic means. TMDD data concepts support the four areas discussed earlier.

As shown in Figure 6, TMDD data concepts include dialogs to start a conversation, messages to request something and receive responses (pertaining to a function or event information or data), and data frames and data elements are used to construct messages.

TMDD has also introduced some flexibility of representing data concepts in two standards-based formats: ASN.1 and XML. The user needs to choose only one format; they cannot be mixed. Also, to achieve interoperability, centers must choose the same format-based data concepts in their system interface specification and design.

Understanding Dialogs

A *dialog* describes a sequence of message exchanges between two entities such as shown in Figure 6 as an external center (EC) talking to an owning center (a TMC) (just as in a conversation between two people).

For example, a request-response dialog would include two messages being shared between an EC and a TMC to accomplish information sharing.

The first message would include the request for information, followed by a message containing the information (response).

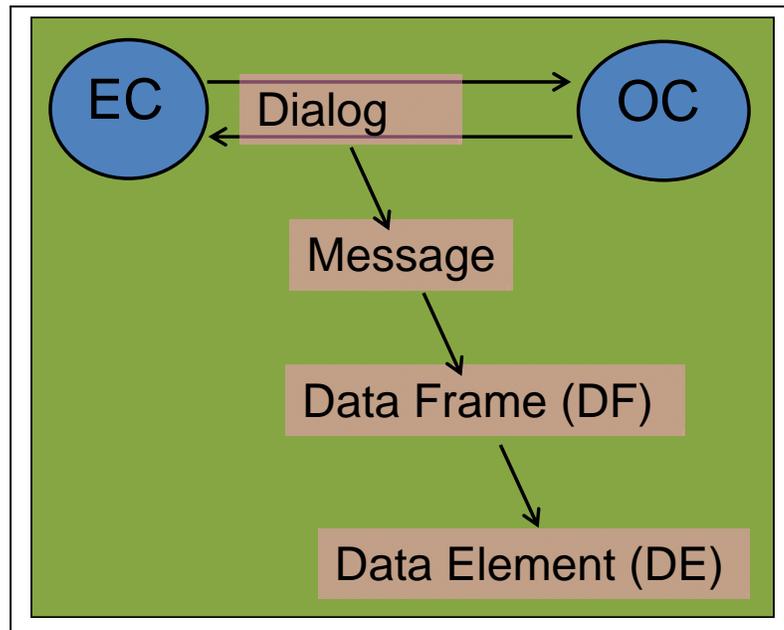


Figure 6: A Representation of the TMDD Data Concepts

Some dialogs are simple and include one or two exchanges, while complex dialogs would include a larger number of steps and alterations of sequence steps based on some criteria (for example, special error handling). Simple dialogs can handle a wide variety of situations or a project may define complex dialogs to meet its special project requirements.

Generic dialogs (Figures 7, 8, and 9) and interface dialogs can be described using the Unified Modeling Language (UML) sequence diagrams as shown below. The examples in this section illustrate how the UML-based diagrams can help in displaying operations and message

inputs/outputs. However, users can devise their own project-specific implementation sequence diagrams using UML.

Message patterns are the building blocks of dialogs. The TMDD standard v3.0 has provided for two basic message patterns:

- **Request-Response.** This message pattern supports the sending of a message followed by a response. For simplicity's sake, this pattern is typically implemented in a purely synchronous fashion, which holds a connection open and waits until the response is delivered or the timeout period expires. (However, request-response may also be implemented asynchronously, with a response being returned at some unknown later time.) Both application-level protocols listed in the previous section support this request-response message pattern, which will be discussed later in this section.
- **Subscription-Publication.** This message pattern supports a subscriber application performing an initial request-response to set up future asynchronous publications from an information publisher application. Both application-level protocols also support the subscription-publication message pattern, which will be discussed later in this section.

The TMDD v3.0 standard has listed three generic dialogs for referencing by the RTM to the 124 customized device classes and other classes dialogs in the RTM; most of these dialogs are designed to facilitate request/response patterns to support operational needs.

Types of Generic Dialogs

1. Request/Response (2.4.1).... Information Sharing
2. Subscription (2.4.2)..... Information Updates
3. Publication (2.4.3)..... Published Information

Requirement ID	Requirement Title	Dialog
3.3.6.3.2.1	Send CCTV Status Information Upon Request	2.4.1
3.3.6.3.2.2	Publish CCTV Status Information	2.4.3
3.3.6.3.2.3	Subscribe to CCTV Status Information	2.4.2

How are Dialogs Referenced in the RTM?

As shown in the referencing table, one of the three generic dialogs is referenced in the project RTM (third column) by the project-selected dialogs (last column in RTM as a standard clause and listed in Section 3 of Volume II). As shown in the table below, an external center had a need to verify CCTV control status, which determined the requirement identified in the second column, which in turn referenced the generic dialog 2.4.1 to carry out the interface dialog 3.1.5.2 listed in Volume II.

Requirement ID (Volume-I)	Requirement title	Dialog See Note Volume II	Data Concept Name	Data Concept Type	Standards Clause Volume II
3.3.6.1.5.1	Send Device Control Status Upon Request	2.4.1	dlDeviceControlStatusRequ	dialog	3.1.5.2

TMDD Generic Request-Response Dialog 2.4.1

Description

The request-response dialog supports the sending of an information or control message initiated by an external center followed by a response by the owner center upon request. Upon error, the owner center returns an error message.

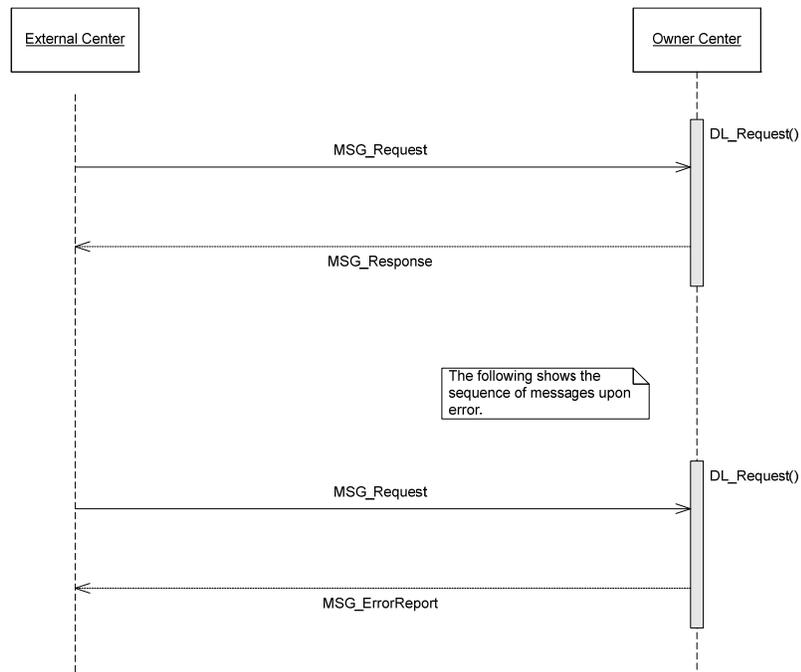


Figure 7: Sequence Diagram of Request/Response Generic Message

TMDD Generic Subscription Dialog 2.4.2

Description

The subscription dialog, initiated by an EC and accepted by an OC, is mandatory for generation of information updates from an OC to an EC. Upon subscription for information updates by an EC, the OC shall provide a confirmation receipt.

Upon error, the OC shall return an error message.

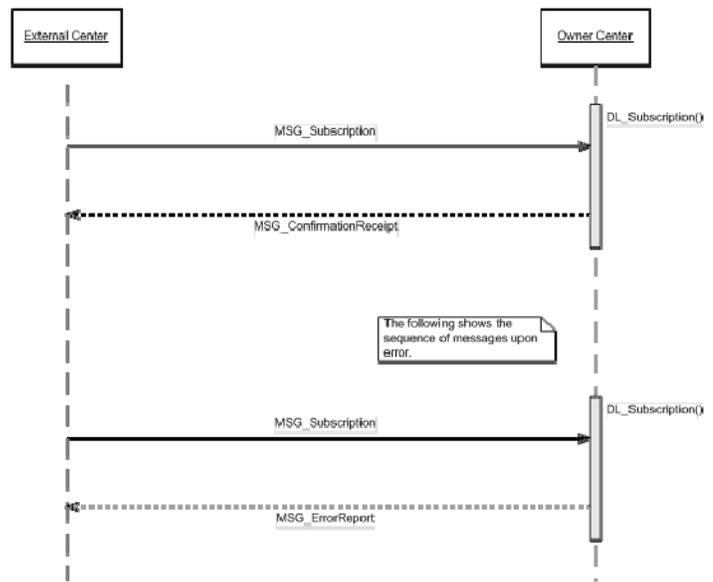


Figure 8: Sequence Diagram of Subscription Generic Message

TMDD Generic Publication Dialog 2.4.3

Description

Upon acceptance of a subscription dialog, an OC shall provide information updates to an EC.
Upon error, the EC shall return an error message.

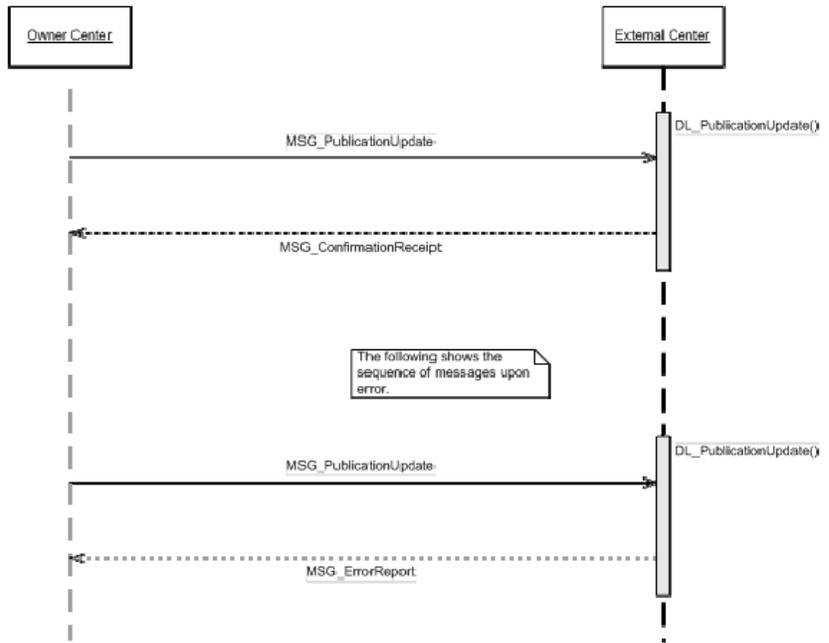


Figure 9: Sequence Diagram of Publication Generic Message

**An Example of How TMDD v3.0 standard Supplied Data Definitions are Used
(Box 1, 2, and 3 Determines “What” We Need Part-Volume I)**

BOX 1: A user (operational need) determines that there is need to share DMS Status and Control to support agency’s traffic management operations
The User is familiar with TMDD v3.0 standard’s two volumes. The user opens the following document. Notice, **ONLY** official copy dated November 12, 2008 obtained from the ITE website is to be utilized for specification preparation. Consult TOC, page 21, Read 2.3.6.4.

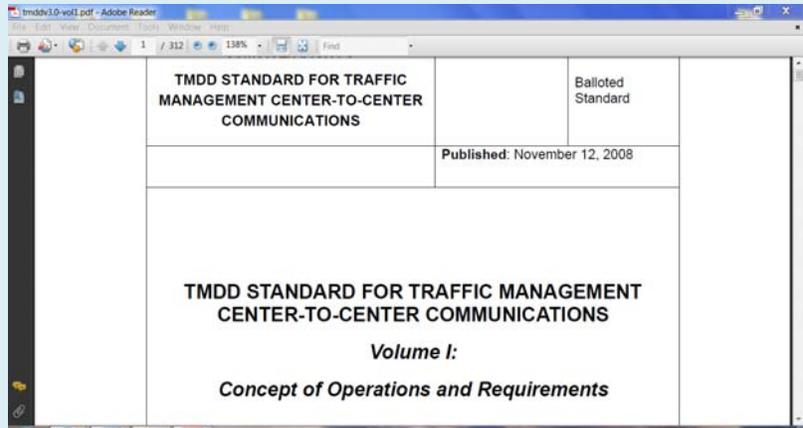
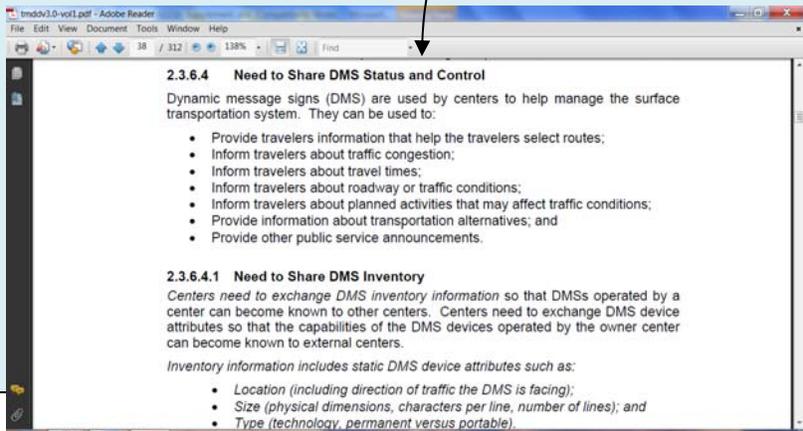
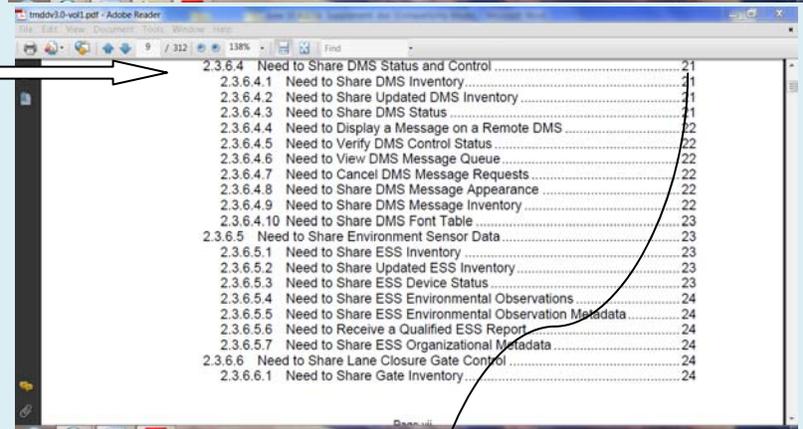


Table of Contents



BOX 2: UN 2.3.6.4 has outlined several User Needs to meet operational needs, pick those that your project needs: As an example, we picked 2.3.6.4.4 after reading definition and mapping to the project on hand.

2.3.6.4.4 Need to Display a Message on a Remote DMS

Centers need to request that a specific message be displayed on a DMS controlled by another center. Messages may be either freeform text messages, in MULTI-string format, or from a library associated with the DMS.

When a control request is received the center that controls the DMS needs to make a determination if the message will be implemented, queued, or rejected. Then, the center that controls the DMS needs to send a response to the center that originated the request describing the status (action taken) on the control request.

After fixing 2.3.6.4.4, a five digit unique UN ID number, both Navigation and Tracing is possible in all TMDD documentation and NRTM and RTM. (Something like using a PRL and RTM concepts in NTCIP device standards). Recall from A202 module, UN ID is unique. UN expresses major capability. YES.

BOX 3: User Needs are ‘Housed’ in NRTM by UN ID Number in the First Column and are associated to Requirements by 4th Column: this is hardwired-fixed.

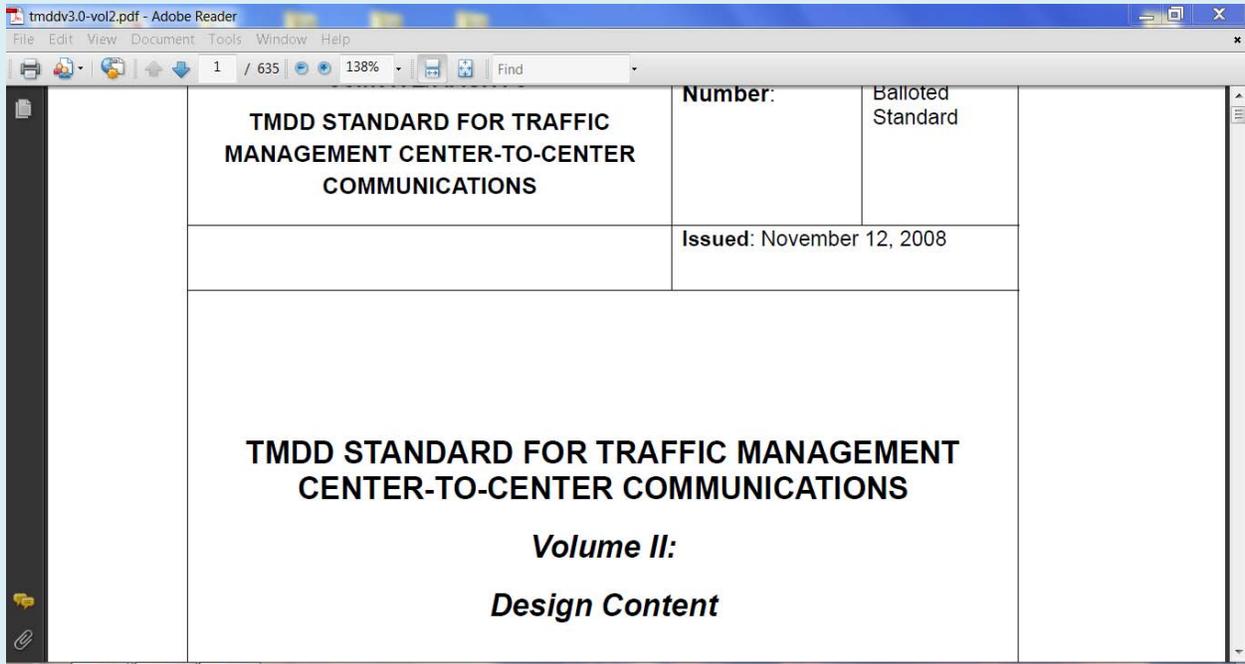
Major Capability: Need to Display a Message on a Remote DMS

UN ID	User Need	UN Selected	Requirement ID	Requirement	Conformance	Support	Other Requirements
2.3.6.4.4	Need to Display a Message on a Remote DMS	Yes / No	3.3.6.1.4.1	Contents of Device Control Request Header	M	Yes	
			3.3.6.1.4.1.1	Required Device Control Request Header Content	M	Yes	
			3.3.6.1.4.1.2.1	Operator Identifier	O	Yes / No	
			3.3.6.1.4.1.2.2	Request Priority	O	Yes / No	
			3.3.6.1.4.1.2.3	Request Expiration Date and Time	O	Yes / No	
			3.3.6.1.4.1.2.4	Event Identifier	O	Yes / No	
			3.3.6.1.4.1.2.5	Event Response Plan	O	Yes / No	
			3.3.6.1.4.1.2.6	External Center Organization	O	Yes / No	
			3.3.6.1.4.1.2.7	Request Date and Time Information	O	Yes / No	
			3.3.6.1.4.2	Contents of Device Control Request Response	M	Yes	
			3.3.6.1.4.2.1	Required Device Control Response Content	M	Yes	
			3.3.6.1.4.2.2.1	Operator Identifier	O	Yes / No	
			3.3.6.1.4.2.2.2	Operator Lock Identifier	O	Yes / No	
			3.3.6.1.4.2.2.3	Owner Center Organization	O	Yes / No	
			3.3.6.1.4.2.2.4	Operator Last Revised Date and Time	O	Yes / No	
			3.3.6.5.3.1	Send DMS Control Response Upon Request	M	Yes	
			3.3.6.5.3.2	Contents of DMS Control Request	M	Yes	
			3.3.6.5.3.2.1	Required DMS Control Request Content	M	Yes	
			3.3.6.5.3.2.2.1	Beacon Control	O	Yes / No	
			3.3.6.5.3.3	Contents of DMS Control Response	M	Yes	

User will select those optional requirements considered needed by selecting YES for conformance, in addition to all Mandatory ones stated above. For example, if a center does not want to know which operator at the other center is making this request, **operator identifier** above is not selected.

NOTE: This window view is taken from the TMDD NRTM on page 224. When you, the user, prepare a project NRTM, the list above will shrink to just a few requirements allocated to this user need. In above view, there 20 requirements, 8 are Mandatory, you are left with 12—you may not need all 12.

BOX 4: Entering Volume II to Use Design Solutions to Meet Requirements



Volume II Design content should lead us to these Dialogs and Messages used in Information Exchanges.

Additional data concepts linked to these messages are also provided but we will not discuss them in detail.

3.1.6	DMS Class Dialogs.....	57
3.1.6.1	DIDMSControlRequest.....	57
3.1.6.2	DIDMSFontTableRequest.....	58
3.1.6.3	DIDMSInventoryRequest.....	59
3.1.6.4	DIDMSInventoryUpdate.....	60
3.1.6.5	DIDMSMessageAppearanceRequest.....	60
3.1.6.6	DIDMSMessageInventoryRequest.....	61
3.1.6.7	DIDMSMessageInventorySubscription.....	62
3.1.6.8	DIDMSMessageInventoryUpdate.....	63
3.1.6.9	DIDMSPriorityQueueRequest.....	64
3.1.6.10	DIDMSStatusRequest.....	65
3.1.6.11	DIDMSStatusUpdate.....	65
3.2.6	DMS Class Messages.....	156
3.2.6.1	DMSControlRequestMsg.....	156
3.2.6.2	DMSFontTableMsg.....	157
3.2.6.3	DMSFontTableRequestMsg.....	158
3.2.6.4	DMSInventoryMsg.....	158
3.2.6.5	DMSMessageAppearanceMsg.....	159
3.2.6.6	DMSMessageAppearanceRequestMsg.....	160
3.2.6.7	DMSMessageInventoryMsg.....	160
3.2.6.8	DMSMessageInventoryRequestMsg.....	161
3.2.6.9	DMSPriorityQueueMsg.....	162
3.2.6.10	DMSStatusMsg.....	162

BOX 5: "USING" the Dialog 3.1.6.1 on page 57, Volume II, to Meet Project Requirements Selected in Box 3 (We have selected XML representation of DCs, but ASN.1 is also available)

3.1.6.1 DIDMSControlRequest

3.1.6.1.1 DEFINITION

A request-response dialog that allows an external center to request an owner center to perform a control action on an owner center's dynamic message sign.

3.1.6.1.2 DIALOG REFERENCE

See Clause 2.4.1 Generic Request-Response Dialog

3.1.6.1.4 XML REPRESENTATION

```
<operation xmlns="http://schemas.xmlsoap.org/wsdl/" name="DiDMSControlRequest">
  <input message="tns:MSG_DMMSControlRequest"/>
  <output message="tns:MSG_DeviceControlResponse"/>
  <fault name="errorReport" message="tns:MSG_ErrorReport"/>
</operation>
```

BOX 6: "USING" the Message 3.2.6.1 on page 156, Volume II, to Meet Project Requirements Selected in Box 3

3.2.6.1 DMMSControlRequestMsg

3.2.6.1.1 DEFINITION

The information content necessary to request a control action of an owner center's dynamic message sign.

3.2.6.1.3 XML REPRESENTATION

```
<xs:element name="dMMSControlRequestMsg" type="DMMSControlRequest"/>
```

BOX 7: Project RTM (Partially filled): we will repeat this until all requirements are included in the project RTM with linked data concepts

Generic dialog 2.4.1 indicates that the dialog is conducted with Request/Response Pattern



Requirement ID	Requirement Title	Dialog	Data Concept Name	Data Concept Type	Standard Clause
3.3.6.1.4.1	Contents of Device Control Request Header		DeviceControlRequestHeader	data-frame	3.3.5.2
3.3.6.1.4.1.1	Required Device Control Request Header Content		OrganizationInformation	data-frame	3.3.17.3
3.3.6.5.3.1	Send DMS Control Response Upon Request	2.4.1	dIDMSControlRequest	dialog	3.1.6.1
3.3.6.5.3.2	Contents of DMS Control Request		dMSControlRequestMsg	message	3.2.6.1
3.3.6.5.3.2.1	Required DMS Control Request Content		DeviceControlRequestHeader	data-frame	3.3.5.2
3.3.6.5.3.2.2.1	Beacon Control		ntcip:DmsMessageBeacon	data-element	NTCIP 1203:5.6.8.6
3.3.6.5.3.3	Contents of DMS Control Response		deviceControlResponseMsg	message	3.2.5.2

Preparing System-Interface Specifications

Introduction

This section expands on the course discussion on how to prepare project specification using the TMDD standard. The section outlines four key steps at the ConOps stage of the SEP and content of the two volumes of the TMDD standard needed to prepare a system interface specification. Please note that a system interface specification is a document that contains complete definitions of the data concepts (dialogs, messages, data frames, and data elements) for the system interface and mapping of the requirements (with the use of RTM).

Specification Considerations

Students may recall detailed discussion in modules A101 and A201 on the acquisition process to procure a system that is based on the ITS standards. Acquisition process documentation includes a complete, consistent, and correct statement on what is desired from the system being procured.

A project specification document (regardless of its title) achieves that purpose in which an agency outlines services desired from a TMDD v3.0 standard-based system interface. This information may be provided in a section in the procurement document.

While the nature of documentation may vary from project to project (based on the type of system being procured) and perhaps agency to agency, in general from the user-needs standpoint, the following components should be considered and included in a specification document:

1. A general background of the project, problem definition, and ConOps/operational needs.
2. A (populated) NRTM for the project: developed by mapping operational needs to those in TMDD.

Additional considerations beyond these two are not discussed here but will be required in a full procurement document. An agency desiring to proceed with acquisition of a system interface must begin with the above steps in consultation with their system support consultant.

The “V” diagram in Figure 10 outlines four key steps for preparing the TMDD-based system interface specification. These steps are occurring in the SEP life cycle. Each step is explained in detail.

Applicable TMDD standard sections are identified in each step and mapped to the stages. Users should be guided by these steps to prepare a project-specific specification. NRTM and RTM tools provided by the TMDD standard must be used.

Step 1: Regional ITS Architecture (See Figure 10)

Readers may recall that the market packages developed by the ITS architecture collect several different subsystems (including equipment packages and terminators) and architecture flows (information flows between subsystems) to provide the desired service. To implement architecture flows between subsystems (centers), the TMDD standard has provided traceability to the National ITS Architecture a selected number of relevant market packages to C2C needs.

As a first step toward preparation for the system interface specification, the reader is advised to also review the work done by the TMDD standard to support communications needs arising from regional ITS architecture *market packages* and *architecture flows*. *Architecture flows* originating from the traffic management center to other centers and the corresponding user needs and requirements are discussed in Volume I, Section 4.

The specification writing process should first check with local regional architecture market package C2C needs and then select appropriate architecture flows and related user needs per Section 4, Volume I. The market packages (partial list) supported by the TMDD standard include *Network Surveillance, Traffic Information Dissemination, Regional Traffic Operations, Traffic Incident Management, Road Weather Data Collection, Roadway Maintenance and Construction, ITS Data Mart, Emergency Call-Taking and Dispatch, Emergency Routing, Disaster Response and Recovery, and Broadcast Traveler Information*. In all cases, the TMDD standard supports not the entire market package but a subset of the interfaces.

Step 2: Selecting User Needs with NRTM

Begin with operational needs: Users should be able to identify potential user needs by observing the C2C operational scenarios. Operational scenarios define the sequence of activities to be performed to satisfy user needs as well as the information flows between entities, both during normal operations and in emergency situations. For example, the operational scenario may include the procedures on how public safety agencies make requests for event information, road network data, device status and inventory, and so forth from a TMC. In a C2C context, the need to communicate with others and/or request and receive information also varies. For example, at some agencies the C2C context may only have a need for sharing DMS messages and/or CCTV control while operating within the freeway environment. At another place, local agencies may be only interested or need the C2C system interface for traffic signals operations.

The TMDD standard lists a broad range of user needs of which local agencies may need only a small subset based on their concept of operations (ConOps). At this step of the SEP, people who will use the intended system interface or will be affected by its use must be engaged in selecting user needs for their specific project. This is critical, because user needs set the tone for the project by clearly defining what will be needed to support an operational problem

solution and dictating how system requirements will emerge in the next phase, during which a system-interfaces design will be done. Only clearly stated user needs using NRTM will ensure that; if users miss them, the result can be an “imperfect” system interface.

Step 3: Tailoring Requirements with NRTM

In the previous step, the first three columns of the NRTM identified and described a unique user need. By doing so, the user had in essence answered the question, “What needs to be done to address a problem in a ConOps?” This was done independently of the notion how it will be done.

In step 3, the last four columns of the same NRTM associated requirements are traced to satisfy that unique need. In the SEP methodology, determination on requirements is critical for system interface design. All system requirements are therefore written in the form of “shall” statements.

Users should be advised that the TMDD standard has traced (allocated) 134 requirements to 125 different user needs. This outcome was a result of a collaborative effort by knowledgeable experts in the field. All requirements were carefully elicited, analyzed, validated, and documented. Most of these requirements are listed as “optional,” allowing users to make the selection for a project. A small number of requirements are determined by the experts to be essential to satisfy certain user needs and are made “mandatory.” To conform to the TMDD standard, mandatory requirements must be included in the specification.

Step 4: Selecting Data Concepts Using RTM

At the high-level design stage in the SEP, we are faced with selecting data concepts, dialogs, messages, data frames, and data elements to complete the system interface specification (this is analogous to selecting building materials for construction work). The TMDD standard provides representation of data concepts in both ASN.1-based and XML-based formats. At this stage, the user must elect one (if he or she has not already done so), and using the RTM as shown in the box, select appropriate data concepts from Volume II.

The sample RTM below illustrates the requirements needed to display a DMS message remotely. The generic dialog (2.4.1) carries out a request/response message pattern for DMS control with two messages. Users should also note that, in a given project, certain requirements may also trace to other ITS standards for data concepts as shown in this example. If such a capability is needed in an implementation, a risk to interoperability could result. In general, adding data concepts from other domain standards not already included in the TMDD could compromise interoperability. Users should take care in such scenarios and prepare accordingly during the implementation process and testing phase.

Mapping the TMDD Standard

The following four steps will guide the user in mapping project-level user needs for specification preparation:

STEP 1: Go to TMDD standard Volume I, Section 4, pages 156–170 of TMDD support to your ITS Architecture market packages, architecture flows.

STEP 2: Using the NRTM (pages 174–295 in the TMDD Standard Volume I, Section 5), select the user needs that address your operational needs. The user need description provided in the ConOps (pages 9–33 in TMDD Standard, Volume 1) will help to better understand the intent and capability of the user needs.

User Needs are located at ConOps stages of the “V” diagram; Requirements follow

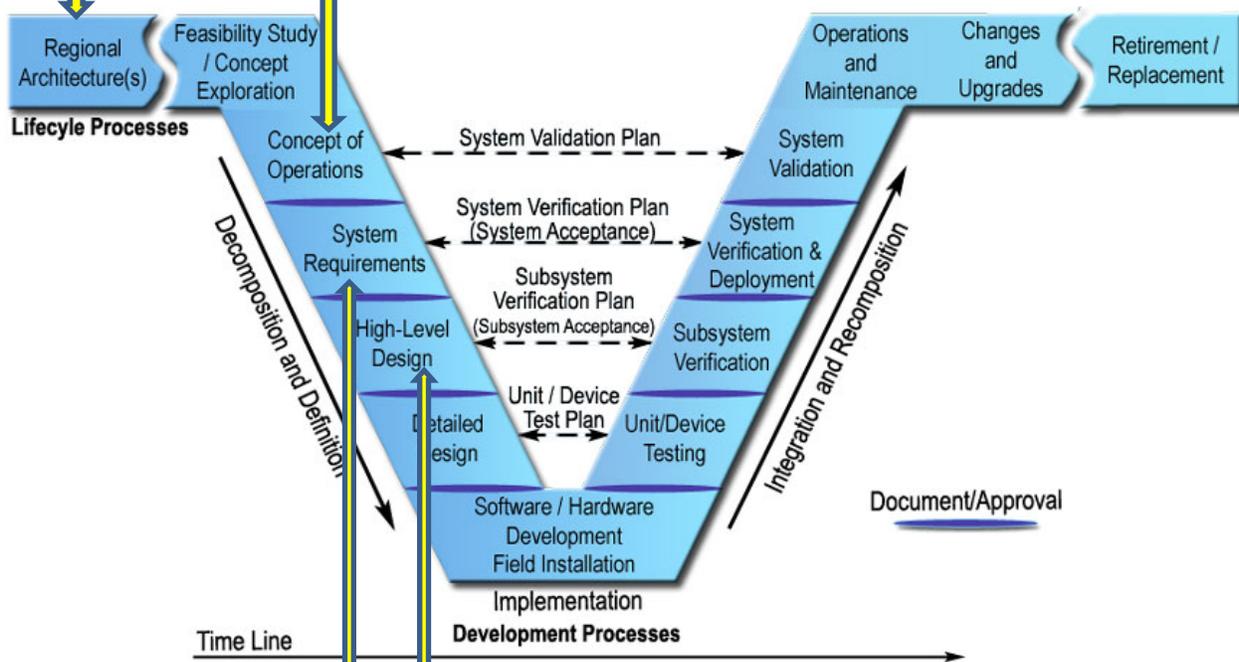


Figure 10: Life Cycle Process

STEP 3: Using the NRTM (pages 174–295 in the TMDD Standard, Volume I), select from the list of associated-requirements those that will satisfy the selected-user needs.

STEP 4: Go to RTM (pages 580–635 in the TMDD Standard, Volume II, Section 4), select data concepts for design elected requirements. This step is often undertaken by a system consultant.

Sample Requirements Traceability Matrix (RTM)

The power of the RTM is apparent from the above discussion. Thus, using the project RTM, there are three benefits: a specification writer can specify which system interface design content is to be implemented in a project specification (not leaving it to guesswork); the system implementer can use the RTM as a vigorous checklist to reduce the risk of failure to conform to the project specification; and, in some cases, a user may benefit from checking with other jurisdictions on potential interoperability issues during separate implementations.

Sample Requirements Traceability Matrix (RTM)

(Exhibit illustrates how requirements (in Volume I) are traced to data concepts (in Volume II) through RTM)

<i>Requirement ID (Volume I)</i>	<i>Requirement title</i>	<i>Dialog See Note Volume II</i>	<i>Data Concept Name</i>	<i>Data Concept Type</i>	<i>Standards Clause See Note Volume II</i>
3.3.6.1.4.2	<i>Contents of Device Control Request Response</i>		<i>deviceControlResponseMsg</i>	<i>message</i>	3.2.5.2
3.3.6.1.4.2.1	<i>Required Device Control Response Content</i>		<i>OrganizationInformation</i>	<i>data-frame</i>	3.3.17.3
3.3.6.1.4.2.1	<i>Required Device Control Response Content</i>		<i>DeviceControlResponse</i>	<i>data-frame</i>	3.3.5.3
3.3.6.1.4.2.1	<i>Required Device Control Response Content</i>		<i>Organization-resource-identifier</i>	<i>data-element</i>	3.4.16.8
3.3.6.1.4.2.1	<i>Required Device Control Response Content</i>		<i>Device-acknowledge-control</i>	<i>data-element</i>	3.4.5.2
3.3.6.1.4.2.2.1	<i>Operator Identifier</i>		<i>Organization-resource-identifier</i>	<i>data-element</i>	3.4.16.8
3.3.6.1.4.2.2.2	<i>Operator Lock Identifier</i>		<i>Organization-resource-identifier</i>	<i>data-element</i>	3.4.16.8
3.3.6.1.4.2.2.3	<i>Owner Center Organization</i>		<i>OrganizationInformation</i>	<i>data-frame</i>	3.3.17.3
3.3.6.1.4.2.2.4	<i>Operator Last Revised Date and Time</i>		<i>Organization-resource-name</i>	<i>data-element</i>	3.4.16.9
3.3.6.1.5.1	<i>Send Device Control Status Upon Request</i>	2.4.1	<i>dIDeviceControlStatusRequest</i>	<i>dialog</i>	3.1.5.2
3.3.6.1.5.2	<i>Contents of the Device Control Status Request</i>		<i>OrganizationInformation</i>	<i>data-frame</i>	3.3.17.3
3.3.6.1.5.2	<i>Contents of the Device Control Status Request</i>		<i>DeviceControlStatusRequest</i>	<i>data-frame</i>	3.3.5.5
3.3.6.1.5.2	<i>Contents of the Device Control Status Request</i>		<i>Organization-resource-identifier</i>	<i>data-element</i>	3.4.16.8
3.3.6.1.5.3	<i>Contents of Device Control Status Response</i>		<i>deviceControlResponseMsg</i>	<i>message</i>	3.2.5.2
3.3.6.3.4	<i>Request CCTV Control Status</i>		<i>deviceControlStatusRequestMsg</i>	<i>message</i>	3.2.5.3

Notes: The **Dialog** column indicates the name of the dialog and represents the highest-level information element defined in Section 2 of Volume II. There are three generic dialogs developed by TMDD (see Volume I, Sections 2.4.1, 2.4.2, and 2.4.3). The **Standards Clause Column** references a clause in either Volume II Section 3 or an external standard containing the definition of the data concept (e.g. command and control of a NTCIP field device such as a DMS will be referenced to the NTCIP standard).

Application-Level Protocols

TMDD is a high level information source, not an interface itself or a communication protocol. As a high level information-level standard, the TMDD is used to develop a system interface, a software entity. As shown in Figure 7, as an information-level data dictionary standard, the TMDD standard defines the content, syntax, and semantics of message exchanges between center-based systems, but it does not define the mechanism of encoding and transporting a message between centers.

NTCIP is a communication application-level protocol designed to transport a message to the other end, independent of the content. The NTCIP family of standards has developed two common protocols:

- NTCIP 2306 C2C-XML is based on the XML data representation and rules of message encoding and transport of the W3C (World Wide Web Consortium) Web services architecture and also provides a way to define dialogs, based on the Web services definition language (WSDL).
- The NTCIP 2304 C2C-DATEX application profile takes the ASN.1 data representation, encodes it with a companion encoding standard (BER), and moves the information from center to center.

As shown in Figure 11, for the traffic management system interface implementation the following standards are required:

1. As a language dictionary: TMDD Standard v3.0
2. As an application profile: NTCIP C2C (One of the two protocols available)

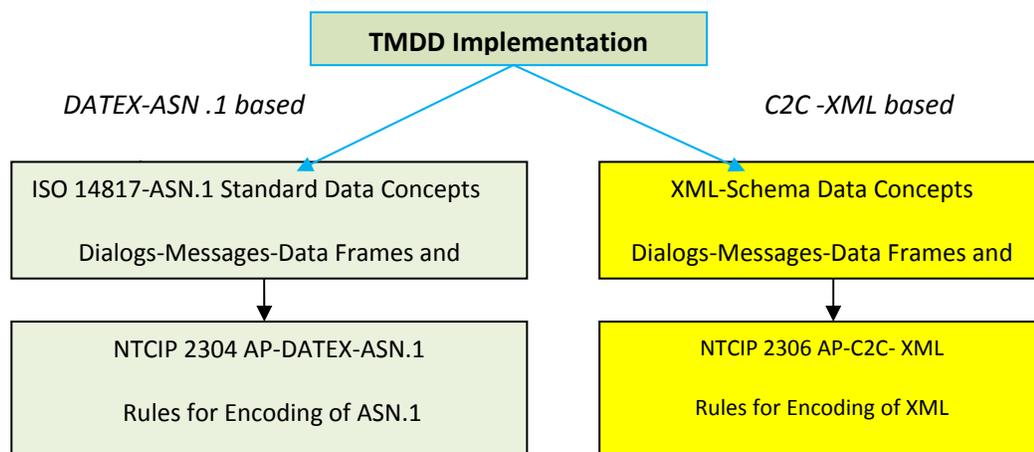


Figure 11: TMDD Implementation

TMDD-Based System Interface Implementation

(Please refer to TMDD v3.0 Guide for details on each setup)

The specification writers desiring to procure a TMDD-based C2C system interface must prepare a project-specific specification using one of the following two types of implementations (note that they cannot be mixed):

C2C XML-based Implementation Requirements

1. *Data concepts: Generic dialogs*
2. *Data concepts: Messages, data frames, and data elements in XML*
3. *Application-level protocol: NTCIP 2306 AP-C2C-XML*
4. *Encoding: XML or SOAP*
5. *Transport: HTTP*
6. *Naming conventions for centers/organizations: NTCIP 1104*
7. *Project-conformance statement*
8. *Other local project requirements*

DATEX-based Implementation Requirements

1. *Data concepts: Generic dialogs*
2. *Data concepts: Messages, data frames, and data elements in ASN.1 representation*
3. *Application-level protocol: NTCIP 2304 AP-DATEX-ASN.1*
4. *Encoding: ASN.1 (or XML)*
5. *Transport: TCP/IP or UDP/IP*
6. *Naming conventions for centers/organizations: NTCIP 1104*
7. *Project-conformance statement*
8. *Other local project requirements*

The specification writers and the system developers should be aware that at the application level, AP-DATEX and AP-C2C XML are not interoperable (they cannot be mixed). The C2C-DATEX is a fixed connection-based approach to information exchange. The C2C XML is a Web services-based (Internet-based) approach to information exchange. Both are distinct in design constructs and therefore only one application-level standard should be chosen. Two or more centers desiring interoperability (an ability to “talk” to each other in real time) must implement a common system interface specification to achieve this objective.

The TMDD v3.0 Guide discusses dialogs, followed by data concepts in ASN.1 representation and data concepts in XML representation and briefly reviews the two application-level protocols used in the TMDD-based C2C system interface implementation.

Conformance Statement

TMDD states the following conformance requirements (Ref. Volume I, Section 1.6, page 6):

The TMDD defines data concepts. The following defines TMDD conformance:

1. To claim conformance to this standard, an implementation shall support all of the *mandatory user needs* and *selected optional needs* defined by the standard.
2. To claim conformance to a user need defined in this standard, an implementation shall conform to all of the *mandatory and selected optional requirements* that trace to the subject user need in the NRTM in Volume 1. This applies to all project user needs, including mandatory and selected optional needs.
3. To claim conformance to a requirement defined in this standard, an implementation shall satisfy the requirement by using all of the dialogs and other data concepts (messages, data frames, and data elements) traced to the subject requirement in the Requirements Traceability Matrix (RTM) in Volume II.

NOTE: The user of this standard is advised that “conformance” to this standard should not be confused with “compliance” to a specification. This standard is as broad as possible to allow a very simple implementation to be “conformant” to this standard. A specification will need to identify the requirements of a particular project and needs to require the support of those requirements. A specification writer is advised to match the requirements of a project with the corresponding standardized requirements defined in this standard to achieve interoperability. This means that functions and requirements defined as “optional” in this standard might need to be selected in a specification (in effect, made “mandatory”).

Off-the-shelf interoperability and interchangeability can only be obtained through well-documented features broadly supported by the industry as a whole. Designing a system that uses features not defined in a standard or not typically deployed in combination with one another will inhibit the goals of interoperability and interchangeability, especially if the documentation of these features is not available for distribution to system integrators.

What if a Requirement is not Found in the TMDD Standard?

Example of a Potential Situation

A hypothetical situation has created a new ConOps that was not addressed by the TMDD and necessitated a definition of a new user need, as well as corresponding requirements and data concepts:

“We have a concept of operations that is unfolding in our region. We are thinking about introducing variable congestion pricing on our high-occupancy vehicle (HOV) facilities and if that happens, a TMC may manage the facilities with a variable pricing scheme imposed by yet another regional center, and they may need to ‘talk’ to each other in real time to communicate pricing schemes. This need is not included in the current standard. What should we do?”

Source: http://www.roadstothefuture.com/Virginia_Freeway_HOV.html

In general, “extensions” to a TMDD-conformant implementation are discouraged because they break interoperability (the reason the TMDD standard was created). However, it is recognized that the TMDD standard does not satisfy every possible user need that can exist between two centers. Therefore, it allows for specific project implementations to “extend” or add new needs, requirements, and data concepts (dialogs, messages, etc.) to the implementation. To support these additional requirements, project implementations are allowed to “extend” the standard by defining new data elements, data frames, or data messages outside the TMDD standard. Please consult your system manager for the project for issues related to interoperability as well.

Extensions (Ref. Volume II, Section 1.6.1)

“The TMDD allows for specific project implementations to “extend” or add new data concepts to the implementation. It is recognized that the standard does not define standardized data concepts for every possible user need that can exist between two centers. Thus, there could be special features or requirements in the implementation that are not supported by the standard. If such features are present, then the systems developer or integrator would need to determine precisely how these features are to be supported without conflicting with the standardized implementations.

“Extensions” to a TMDD-compliant implementation are discouraged because they break interoperability. However, the standards organizations recognize the need to satisfy functional requirements not supported by this standard. To support these additional requirements, project implementations are allowed to “extend” the standard by defining new data elements, data frames, or data messages outside this TMDD standard.

To maintain conformance to this standard, the following rules for “extending” the TMDD standard must be met:

- All functional requirements already supported by this standard must be implemented as defined by the standard. An implementation may NOT define a new data element or data message if that functional requirement is already supported by this standard. In other words, the implementation may NOT completely replace a partially incomplete feature of the standard with a complete custom feature.
- An implementation may add new data elements beyond those data elements defined by the standard. However, an extension cannot reuse an existing name or identifier already defined by the standard.
- Extending a data element to support additional enumerations is allowed if the enumerations are clearly and uniquely defined.
- Extending the range of an existing data element requires that the data element be renamed.
- If an implementation has a different interpretation of the meaning of a data element or how the data element is to be used as defined by this standard, a new data element is to be created.
- If a new data element is added to a message defined by this standard, that data element shall be marked as OPTIONAL. An implementation may decide that the data element is mandatory for all centers participating in the implementation, but the data element will still be tagged as an OPTIONAL element to maintain compatibility with other implementations.
- Any extensions shall be documented by the owning agency or agencies and/or the systems integrator, either in the XML schema or in ASN.1 notation.
- If extensions are made to a message defined by the standard, whether through the addition of data elements to the message or via changes to an existing data element (with the exception of additional enumerations to a data element), that data message shall be renamed to prevent confusion or ambiguity for the purposes of interoperability.
- An implementation must ignore any attributes or elements in a valid MIB, schema, or whatever format that it does not recognize. The must-ignore rule applies only to the attribute or element and does not apply for any descendants of the attribute or element.”

References

1. **TMDD V3.0 Standard Documentation** (Volume I and II) available at:
 - a. <http://www.ite.org/standards/TMDD/>
 - b. To download PDF files of the TMDD v3.0 Standard Volume I and II, click on the shopping cart symbol in the last column at <http://distribution.ite.org/process.asp>
2. ISO/FDIS 14817:2002(E), *Transport information and control systems — Requirements for an ITS/TICS central data registry and ITS/TICS data dictionaries*, v2.0,2007. http://www.iso.org/iso/catalogue_detail?csnumber=36030
3. *IEEE Standard Glossary of Software Engineering Terminology*, IEEE 610.12-1990, <http://standards.ieee.org/findstds/standard/610-1990.html>

Subject Area Guides

4. *Guide to Traffic Management Data Dictionary (TMDD) Standard v3.0 for Traffic Management Center-to-Center Communications*, July 12, 2011 published). <http://www.ite.org/standards/TMDD/>
5. NTCIP Guide, Information Report 9001, www.ntcip.org

Center to Center Case Studies (Deployments)

6. AZ Tech C2C Interface Specification and ConOps Plan www.aztech.org/docs/c2c/az-c2c-spec.pdf
7. Developing and Using a Concept of Operations in Transportation Management Systems-FHWA, 2005, Handbook tmcpfs.ops.fhwa.dot.gov/cfprojects/uploaded.../conops_tms_handbook.pdf
8. Deschutes County ITS Plan, Oregon DOT, User Needs Assessment by DKS Associates, Inc.
9. AZTech Center-to-Center (C2C) Interface Specification (Version 1.1), Prepared by Consensus Systems Technologies Corp., Maricopa County Department of Transportation, Arizona Department Transportation www.aztech.org/docs/c2c/az-c2c-spec.pdf

Systems Engineering Guides/Standards Information

10. *Systems Engineering for ITS—An Introduction for Transportation Professionals*, FHWA: <http://ops.fhwa.dot.gov/publications/seitsguide/seguide.pdf>
11. *Systems Engineering Guidebook*, Caltrans and FHWA, February 2005.
12. http://www.dot.ca.gov/newtech/docs/se_guidebook_ver1-12_14_05.pdf
13. *Systems Engineering Handbook*, International Council on Systems Engineering (INCOSE), v3.2, 2010. www.incose.org/ProductsPubs/products/sehandbook.aspx
14. USDOT, RITA, ITS- JPO, www.its.dot.gov/index.htm (Fact sheets)
15. USDOT Standards Program, www.standards.its.dot.gov
16. USDOT, FHWA, Freeway Operations and Management http://ops.fhwa.dot.gov/freewaymgmt/frwy_ops.htm