

## **Appendix K**

### **New York State ITS Standards Specification Development Guide**

# **Example New York State ITS Project Information and Application Level Center-to-Center (C2C) Interface Specification**

**Prepared for**

**New York State Department of Transportation**

**Prepared by**

**Consensus Systems Technologies Corp.**

**November 22, 2006**

## Table of Contents

1	Introduction.....	1
1.1	Project Background.....	1
1.2	Specification Development Approach .....	1
1.3	Document Organization .....	2
2	General Requirements.....	3
2.1	References .....	3
2.2	Conformance Statement .....	3
2.3	Glossary of Terms .....	4
3	Application Profile for Center-to-Center Communications and PICS .....	6
4	Center Interface Dialog Definitions.....	8
4.1	Dialogs and WSDL Operations.....	8
4.1.1	Dialog – DL_DMSTControlRequest.....	8
4.1.2	Dialog – DL_DMSTInventoryRequest .....	9
4.1.3	Dialog – DL_DMSTInventoryUpdateSubscriptionPublication .....	10
4.1.4	Dialog – DL_DMSTStatusRequest .....	11
4.1.5	Dialog – DL_DMSTStatusUpdateSubscriptionPublication .....	12
4.1.6	Dialog – DL_CancelSubscriptionRequest .....	13
4.1.7	Dialog – DL_DeviceCancelControlRequest.....	14
4.2	Web Services Description Language Worksheet .....	14
5	Center Interface Message, Data Frame, and Data Element Definitions .....	17
5.1	#1 AuthorizationSet (Data Frame).....	17
5.2	#2 BroadcastAlerts (Data Frame).....	18
5.3	#3 BroadcastAlertsItem (Data Element) .....	18
5.4	#4 C2C Message Publication (Data Frame).....	18
5.5	#5 C2C Message Subscription (Data Frame).....	19
5.6	#6 C2C_MessageReceipt (Data Frame).....	19
5.7	#7 C2C_SubscriptionTimeFrame (Data Frame) .....	20
5.8	#8 CancelSubscriptionRequest (Message).....	20
5.9	#9 CONTACT_EmailAddress_text (Data Element).....	20
5.10	#10 CONTACT_Identifier_identifier (Data Element) .....	21
5.11	#11 CONTACT_PersonName_text (Data Element) .....	21
5.12	#12 CONTACT_PersonTitle_text (Data Element) .....	21
5.13	#13 CONTACT_PhoneAlternate_text (Data Element).....	21
5.14	#14 CONTACT_PhoneNumber_text (Data Element).....	22
5.15	#15 ContactDetails (Data Frame) .....	22
5.16	#16 DateTimePair (Data Frame) .....	22
5.17	#17 DEVICE_AcknowledgeControl_code (Data Element) .....	23
5.18	#18 DEVICE_CommandEndTime_utc (Data Element).....	24
5.19	#19 DEVICE_CommandRequestPriority_number (Data Element) .....	24
5.20	#20 DEVICE_Identifier_identifier (Data Element) .....	24
5.21	#21 DEVICE_OperationalStatus_code (Data Element).....	24
5.22	#22 DEVICE_RequestIdentifier_identifier (Data Element).....	25
5.23	#23 DEVICE_Type_code (Data Element) .....	25

5.24	#24 DeviceArchiveStatusSubscription (Message)	26
5.25	#25 DeviceCancelControlRequest (Message)	27
5.26	#26 DeviceControlSubscriptionResponse (Message)	27
5.27	#27 DeviceInventoryRequest (Message)	28
5.28	#28 DeviceInventorySubscription (Message)	29
5.29	#29 DeviceStatusRequest (Message)	29
5.30	#30 DeviceStatusSubscription (Message)	30
5.31	#31 DMSControlRequest (Message)	30
5.32	#32 DMSDeviceStatus (Data Frame)	31
5.33	#33 DMSInventoryPublication (Message)	31
5.34	#34 DMSInventoryResponse (Message)	32
5.35	#35 DMSStatusPublication (Message)	32
5.36	#36 DMSStatusResponse (Message)	33
5.37	#37 EXT_ATIS_Date (Data Element)	33
5.38	#38 EXT_ATIS_Time (Data Element)	33
5.39	#39 EXT_DmsMessageBeacon (Data Element)	34
5.40	#40 EXT_DmsMessageMultiString (Data Element)	34
5.41	#41 EXT_DmsMessageNumber (Data Element)	35
5.42	#42 EXT_DmsMessageTimeRemaining (Data Element)	35
5.43	#43 EXT_DmsSignTechnology (Data Element)	35
5.44	#44 EXT_LRMS_Distance (Data Frame)	36
5.45	#45 EXT_LRMS_Height (Data Frame)	39
5.46	#46 EXT_LRMS_HorizontalDatum (Data Element)	39
5.47	#47 EXT_LRMS_Latitude (Data Element)	40
5.48	#48 EXT_LRMS_Longitude (Data Element)	40
5.49	#49 EXT_LRMS_VerticalDatum (Data Element)	41
5.50	#50 EXT_LRMS_VerticalLevel (Data Element)	41
5.51	#51 Filters (Data Frame)	41
5.52	#52 FreeText (Data Element)	42
5.53	#53 InformationalText (Data Element)	42
5.54	#54 LINK_Direction_code (Data Element)	42
5.55	#55 LINK_RouteDesignator_identifier (Data Element)	43
5.56	#56 ORGANIZATION_CenterIdentifier_identifier (Data Element)	44
5.57	#57 ORGANIZATION_Identifier_identifier (Data Element)	44
5.58	#58 ORGANIZATION_Name_text (Data Element)	44
5.59	#59 ORGANIZATION_SubOrganizationName_text (Data Element)	44
5.60	#60 OrganizationInformationLong (Data Frame)	45
5.61	#61 OrganizationInformationShort (Data Frame)	45
5.62	#62 ReturnAddress (Data Element)	46
5.63	#63 SECURITY_Password_text (Data Element)	46
5.64	#64 SECURITY_UserName_text (Data Element)	46
5.65	#65 SubscriptionAction (Data Frame)	46
5.66	#66 SubscriptionActionItem (Data Element)	47
5.67	#67 SubscriptionCount (Data Element)	47
5.68	#68 SubscriptionFrequency (Data Element)	47
5.69	#69 SubscriptionID (Data Element)	48

5.70	#70 SubscriptionName (Data Element).....	48
5.71	#71 SubscriptionType (Data Frame) .....	48
5.72	#72 SubscriptionTypeItem (Data Element).....	49
5.73	#73 TimeInterval (Data Element).....	49
Appendix A	Requirements Traceability Matrix	
Appendix B	Web Services Description Language Listing	
Appendix C	XML Schema Listing	

## Revision History

Filename	Version	Date	Author	Comment
NYStateSpecGuide_ApK.doc	0.1	10/08/06	M. Insignares	Initial Draft
NYStateSpecGuide_ApK.doc	0.3	11/22/06	M. Insignares	Updated formatting of appendices in the specification guide.

# 1 Introduction

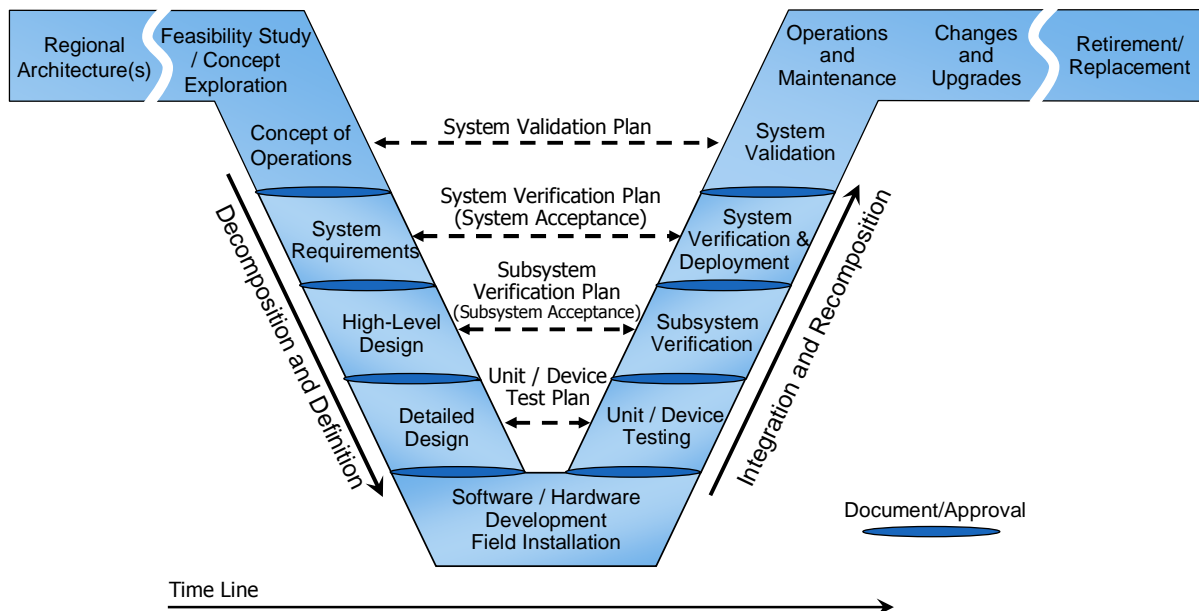
## 1.1 Project Background

The New York State Department of Transportation (NYSDOT) has initiated a project to deploy an end-to-end communications system to improve transportation information dissemination to motorists and neighboring agencies. This deployment will result in the expansion of the NYSDOT ITS program, and work towards the deployment of transportation operations communications network.

This (Draft) NY State Center-to-Center Specification has been developed through the consensus input of regional stakeholders to standardize the communication interfaces of center systems. Only the center's interface (the way the center appears to other centers) is being standardized, the center system software, design, algorithms and methodologies that implement the interface, provided by numerous system integrators and vendors, are not covered by this specification.

## 1.2 Specification Development Approach

Following a Systems Engineering Methodology, a user-needs assessment and concept of operations (ConOps) were developed. Based on the ConOps, functional requirements and stakeholder agreement were developed. The figure below shows the application of the 'VEE' model to the development of the NY State C2C Interface Specification Development.



The system functional requirements for NY State DMS (Dynamic Message Sign) System are the basis for development of the technical specifications for information exchange among centers in the region.

Stepping from problem (requirements) to solution (design) a requirements traceability matrix (RTM) was developed to document how the analysis of requirements translates to the project-specific design of center interfaces. The RTM is shown in Appendix A.

### **1.3 Document Organization**

This document is comprised of five Sections and three Appendices.

- Section 1 - Introduction. This section provides an overview of the document and introductory material.
- Section 2 - General Requirements. Contains a list of references to other documents, a conformance statement, and a glossary of terms.
- Section 3 - Application Profile and Transport. Addresses the application level and transport standards for this specification.
- Section 4 - Center Interface Dialog Definitions. Contains a listing of required dialogs and messages and a mapping of the dialogs and messages to a WSDL worksheet.
- Section 5 - Center Interface Message, Data Frame, and Data Element Definitions. Contains a description of all messages, data frames, and data elements.
- Appendix A - Requirements Traceability Matrix. Shows the RTM in table form.
- Appendix B - Web Services Description Language Listing.
- Appendix C - XML Schema Listing.

## 2 General Requirements

This specification is based on project and ITS standards information contained in other documents. All the documents listed in **Table 2.1** are available on the project web site. The documents and standards referenced provide background information necessary to understand this specification's design content. The RTM provides a bridge between the NYState ITS Project functional requirements and the design elements contained in this specification.

### 2.1 References

**Table 2-1. References**

Organization	Document Title	Date
NY State	NY State ITS Project Center-to-Center Needs Assessment and Concept of Operations	May 27, 2005
NY State	NY State ITS Project Center-to-Center Dynamic Message Sign (DMS) Systems (TMS) Functional Requirements	January 3, 2006
NTCIP	NTCIP 2306 v01.68 - Application Profile for XML Message Encoding and Transport in ITS Center-to-Center Communications	February 8, 2006
NTCIP	NTCIP 9010 v01.09 - Information Report, XML in Center-to-Center Communications	November 23, 2004
ITE/AASHTO	Traffic Management Center-to-Center Communications {Advanced Traffic Management Data Dictionary (TMDD) and Message Sets (MS)} version 2.1	June 30, 2004

### 2.2 Conformance Statement

A center's interface conforms with this specification as follows:

1. A WSDL document and XML Schema exist that conform with the WSDL contained in the Appendix. Each implementing center must replace the WSDL Service EndPoint (contained in the service section of the WSDL) with an appropriate URI. The WSDL SoapAction named "dummy.cgi" (contained in the binding section of the WSDL) must be replaced with an appropriate EndPoint suffix, OR replaced by the following string "" (double-quote single-quote space single-quote double-quote), which indicates a blank SoapAction. The URL designation of the SOAP EndPoint of an implementing center is the concatenation of the service and binding section EndPoints.
2. A center system shall demonstrate the support of the web services described in the WSDL and XML Schema by: 1) XML message inputs and outputs described in the WSDL can be reproduced under a system test, 2) XML messages validate against the referenced schema, 3) messages can be exchanged using the dialogs defined in the WSDL.
3. XML Messages are encoded using the SOAP.
4. Soap EndPoints are accessible via HTTP.



5. The transport for HTTP is TCP/IP and the SOAP EndPoint may be accessed via an IP PING and TCP traceroute.

## **2.3 Glossary of Terms**

The following glossary defines terms and acronyms used in this standard.

**Table 2-2. Glossary of Terms**

<b>Term</b>	<b>Definition</b>
<b>AASHTO</b>	American Association of State Highway and Transportation Officials
<b>AP</b>	Application Profile
<b>C2C</b>	Center-to-Center
<b>FTP</b>	File Transfer Protocol, a file transfer protocol of the IETF.
<b>GZIP</b>	A file compression standard of the IETF.
<b>HTTP</b>	Hypertext Transfer Protocol
<b>HTTPS</b>	Secure HTTP (Hypertext Transfer Protocol)
<b>IETF</b>	Internet Engineering Task Force
<b>IP</b>	Internet Protocol
<b>ITE</b>	Institute of Transportation Engineers
<b>ITIS</b>	International Traveler Information System (SAE J2540-1)
<b>ITS</b>	Intelligent Transportation Systems
<b>LRMS</b>	Location Referencing Message Specification
<b>MIME</b>	Multipurpose Internet Mail Extensions (IETF RFC 1521)
<b>MS/ETMCC</b>	Message Sets for External Traffic Management Center Communications
<b>NEMA</b>	National Electrical Manufacturers Association
<b>NTCIP</b>	National Transportation Communications for ITS Protocol
<b>NTCIP 9010</b>	NTCIP Information Report on ITS Center-to-Center Communications
<b>PICS</b>	Profile Implementation Conformance Statement
<b>PN</b>	Profile Need
<b>PR</b>	Profile Requirement
<b>PRL</b>	Profile Requirements List

Term	Definition
<b>SAE</b>	Society of Automotive Engineers
<b>SOAP</b>	Simple Object Access Protocol
<b>TCP</b>	Transmission Control Protocol
<b>TMDD</b>	Traffic Management Data Dictionary, a joint standard of ITE/AASHTO
<b>W3C</b>	World Wide Web Consortium
<b>WSDL</b>	Web Services Description Language
<b>XML</b>	eXtensible Markup Language

### 3 Application Profile for Center-to-Center Communications and PICS

The NY State C2C Interface Specification shall use the Application Profile for XML Message Encoding and Transport for ITS Center to Center Communications, NTCIP 2306 (NTCIP C2C XML). NY State centers shall communicate XML messages using the SOAP encoding of XML over the HTTP protocol, and both the request-response and subscription-publication message patterns.

A Project Implementation Conformance Statement (PICS) (taken directly from the 2306 standard) for the project is shown below.

**Table 3-1. NTCIP 2306 PICS (Profile Implementation Conformance Statement)**

Profile Requirements List (PRL)		NTCIP 2306 Section	NTCIP 2306 Mandatory / Optional	NTCIP 2306 Profile Requirement	Project Requirement
<b>1.0</b>	<b>SOAP over HTTP</b>				
	<b>a) WSDL Request-Response</b>		<b>M</b>		<b>Y</b>
	- WSDL General	6.1	M		Y
	- Definitions	6.2	M	PR 3.1	Y
	- Types/Schema	6.3	M	PR 3.1, 3.2	Y
	- Message	6.4	M	PR 3.3	Y
	- PortType (Interfaces)	7.1.1	M	PR 4.1.1	Y
	- Binding (Transport)	7.1.2	M	PR 4.1.2a	Y
	- Service (Transport)	7.1.3	M	PR 4.1.2a	Y
	<b>b) WSDL Publish-Subscribe</b>		<b>M</b>		<b>Y</b>
	- WSDL General	6.1	M		Y
	- Definitions	6.2	M	PR 3.1	Y
	- Types/Schema	6.3	M	PR 3.1, 3.2	Y
	- Message	6.4	M	PR 3.3	Y
	- PortType (Interfaces)	6.5, 7.2.1	M	PR 4.2.1	Y
	- Binding (Transport)	7.2.2	M	PR 4.2.2a	Y
	- Service (Transport)	7.2.3	M	PR 4.2.2a	Y
	<b>c) Message Encoding</b>				<b>Y</b>
	SOAP	4.2.2	M	PR 1.2, 4.2.1a, 4.2.1b	Y
	<b>d) Message Transport</b>				<b>Y</b>
	HTTP	5.1.3	M	PR 2.1a, 4.1.2a, 4.2.2a	Y
	HTTPS	5.1.4, 6.6	O	PR 2.1b	Y
<b>2.0</b>	<b>XML over HTTP</b>				
	<b>a) WSDL Request Only (XML Direct)</b>		<b>O</b>		<b>N</b>
	- WSDL General	6.1	M		N

<b>Profile Requirements List (PRL)</b>		<b>NTCIP 2306 Section</b>	<b>NTCIP 2306 Mandatory / Optional</b>	<b>NTCIP 2306 Profile Requirement</b>	<b>Project Requirement</b>
	- Definitions	6.2	M	PR 3.1	N
	- Types/Schema	6.3	M	PR 3.1, 3.2	N
	- Message	6.4	M	PR 3.3	N
	- PortType (Interfaces)	8.1.1	M	PR 4.4.1, 3.4	N
	- Binding (Transport)	8.1.2	M	PR 4.4.2a	N
	- Service (Transport)	8.3	M	PR 4.4.2a	N
	<b>b) WSDL Request-Response</b>		<b>O</b>		<b>N</b>
	- WSDL General	6.1	M		N
	- Definitions	6.2	M	PR 3.1	N
	- Types/Schema	6.3	M	PR 3.1, 3.2	N
	- Message	6.4	M	PR 3.3	N
	- PortType (Interfaces)	8.2.1	M	PR 4.1.1	N
	- Binding (Transport)	8.2.2	M	PR 4.1.2b	N
	- Service (Transport)	8.3	M	PR 4.1.2b	N
	<b>d) Message Encoding</b>				<b>N</b>
	XML Text	4.1.2	M	PR 1.1a, 4.2.2b	N
	XML Gzip	4.1.2	O	PR 1.1 a, 4.2.2b	N
	<b>e) Message Transport</b>				<b>N</b>
	HTTP	5.1.1, 5.1.2	M	PR 2.1a, 4.2.2b	N
	HTTPS	5.1.4, 6.6	O	PR 2.1a, 4.2.2b	N
<b>3.0</b>	<b>XML over FTP</b>				<b>N</b>
	<b>a) WSDL Request Only (XML Direct)</b>		<b>O</b>		N
	- WSDL General	6.1	M		N
	- Definitions	6.2	M	PR 3.1	N
	- Types/Schema	6.3	M	PR 3.1, 3.2	N
	- Message	6.4	M	PR 3.3	N
	- PortType (Interfaces)	9.1.2	M	PR 4.4.1, 3.4	N
	- Binding (Transport)	9.1.3	M	PR 4.4.2b	N
	- Service (Transport)	9.1.4	M	PR 4.4.2b	N
	<b>b) Message Encoding (one of the following)</b>				<b>N</b>
	XML Text	4.1.1	O	PR 1.1a, 4.4.2b	N
	XML Gzip	4.1.2	O	PR 1.1b, 4.4.2b	N
	<b>c) Message Transport</b>				<b>N</b>
	FTP	5.2.1	M	PR 2.2a, 4.4.2b	N

## 4 Center Interface Dialog Definitions

### 4.1 Dialogs and WSDL Operations

This section contains a series of UML diagrams that illustrate the operations, message inputs and outputs of the NY State C2C information exchange dialogs. Each diagram shows a 'Center' and 'Other Center'. The 'Other Center' represents a requester / subscriber, while the 'Center' is the responder / publisher.

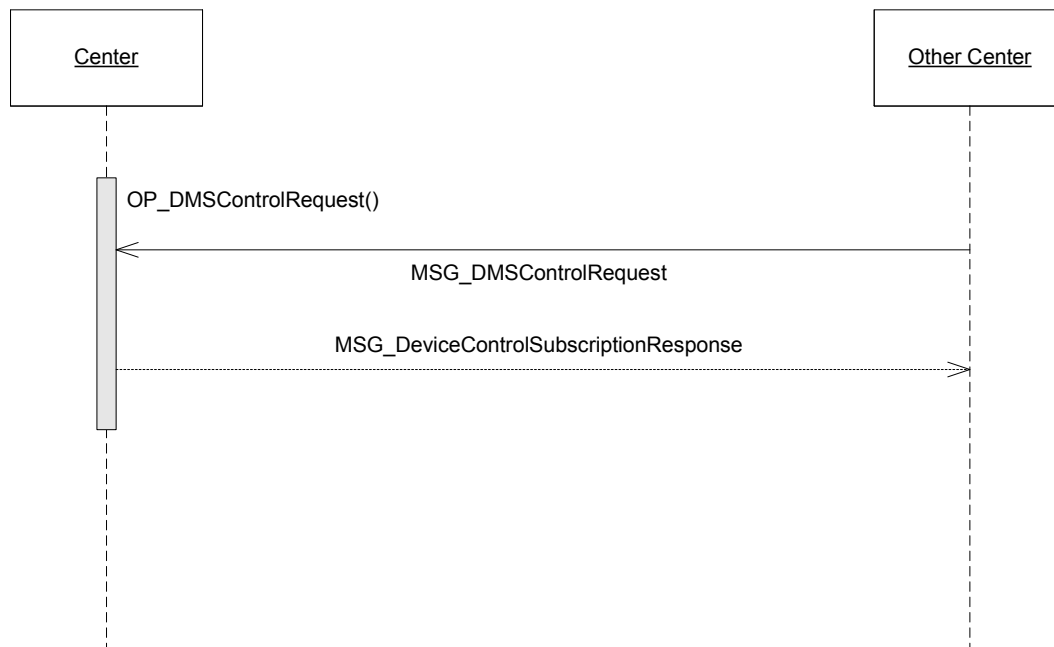
WSDL Operations are shown next to the grey vertical box on the UML lifeline and have the prefix 'OP\_'. Messages that represent the inputs and outputs to the operation (also read as center function), start with the 'MSG\_' prefix.

For a complete description of the system functional requirements associated with each dialog, please consult the Requirements Traceability Matrix contained in the Appendix.

#### 4.1.1 Dialog – DL\_DMSTControlRequest

The figure below illustrates the operations, message inputs and outputs of the DMSTControlRequest Dialog.

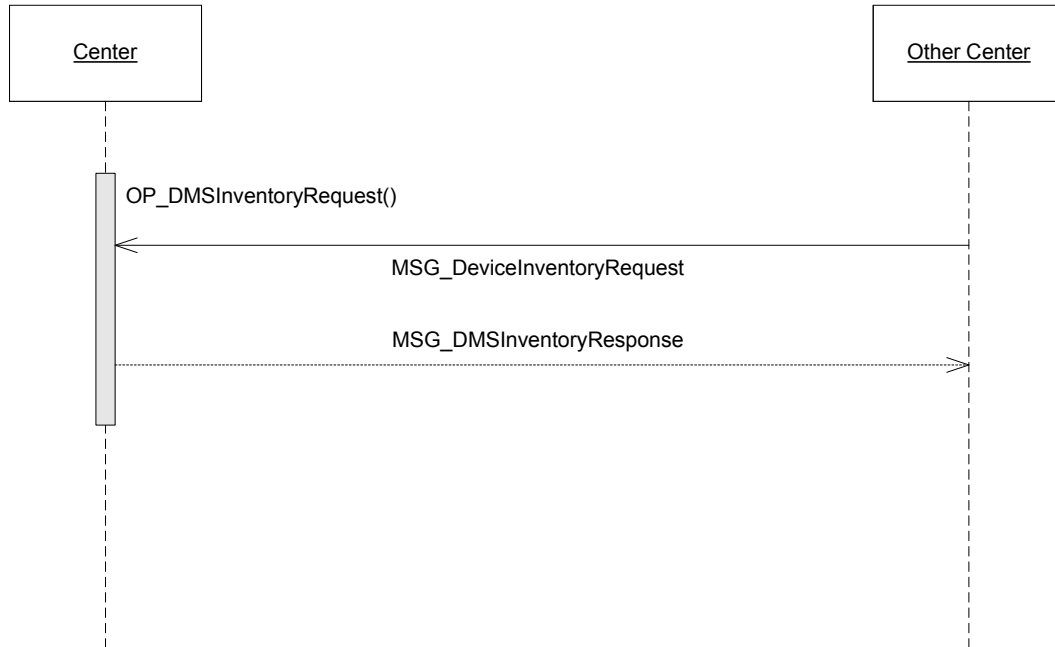
Figure 4-1. DL\_DMSTControlRequest



#### 4.1.2 Dialog – DL\_DMSInventoryRequest

The figure below illustrates the operations, message inputs and outputs of the DMSInventoryRequest Dialog.

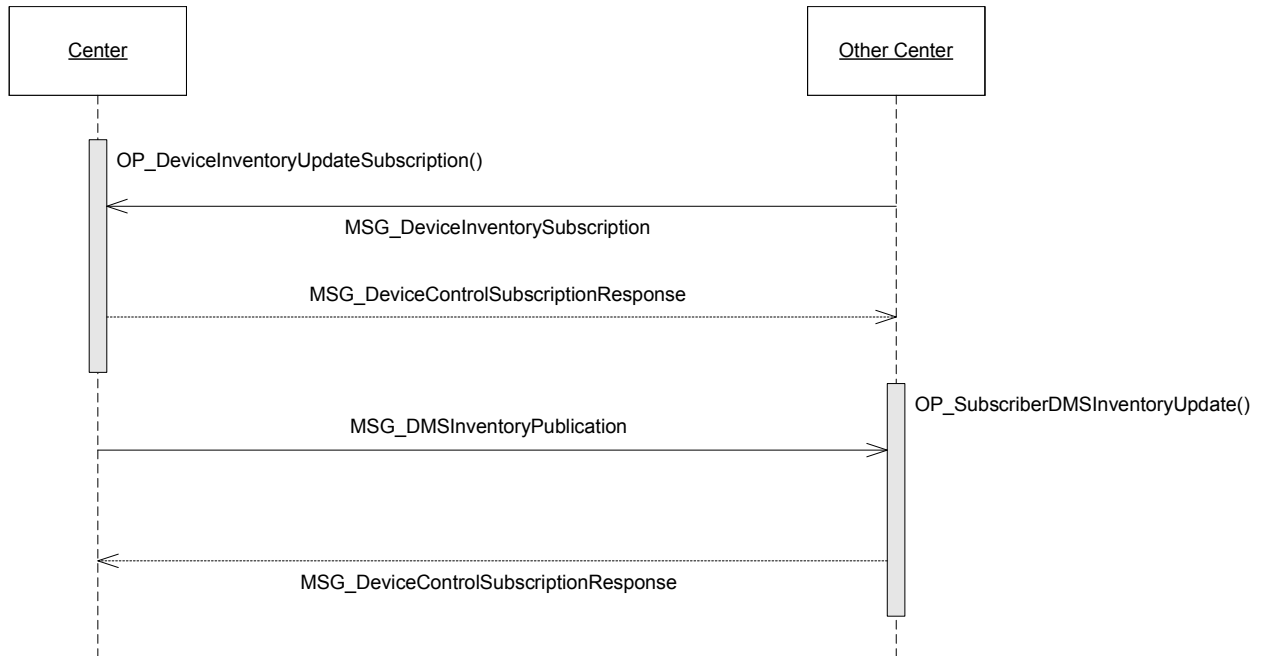
**Figure 4-2. DL\_DMSInventoryRequest**



### 4.1.3 Dialog – DL\_DMSInventoryUpdateSubscriptionPublication

The figure below illustrates the operations, message inputs and outputs of the DMSInventoryUpdateSubscriptionPublication Dialog.

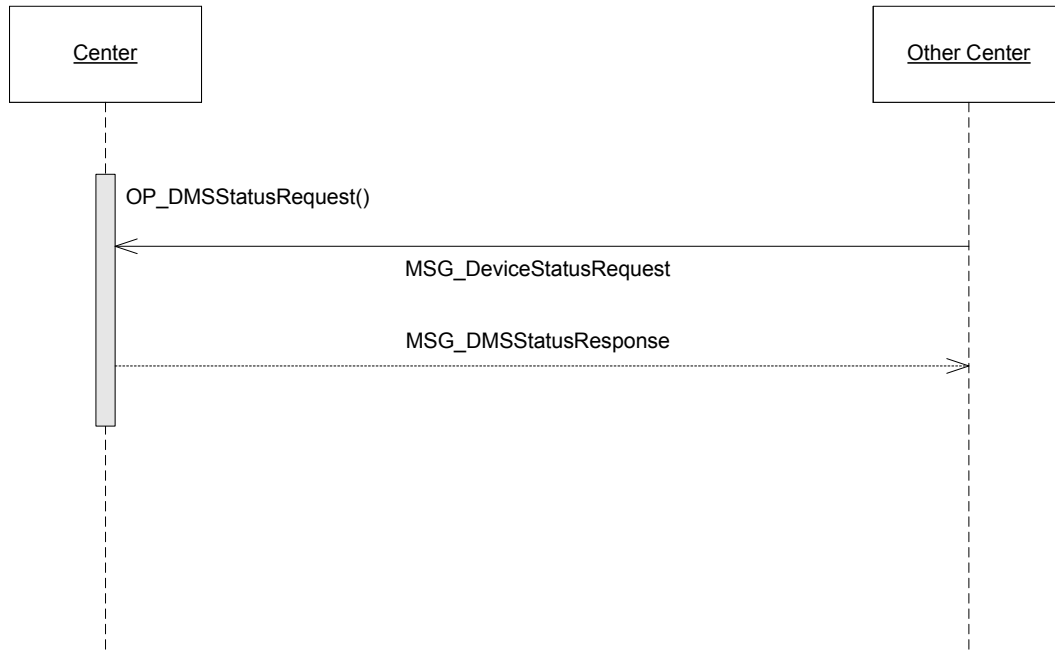
Figure 4-3. DL\_DMSInventoryUpdateSubscriptionPublication



#### 4.1.4 Dialog – DL\_DMSStatusRequest

The figure below illustrates the operations, message inputs and outputs of the DMSStatusRequest Dialog.

**Figure 4-4. DL\_DMSStatusRequest**

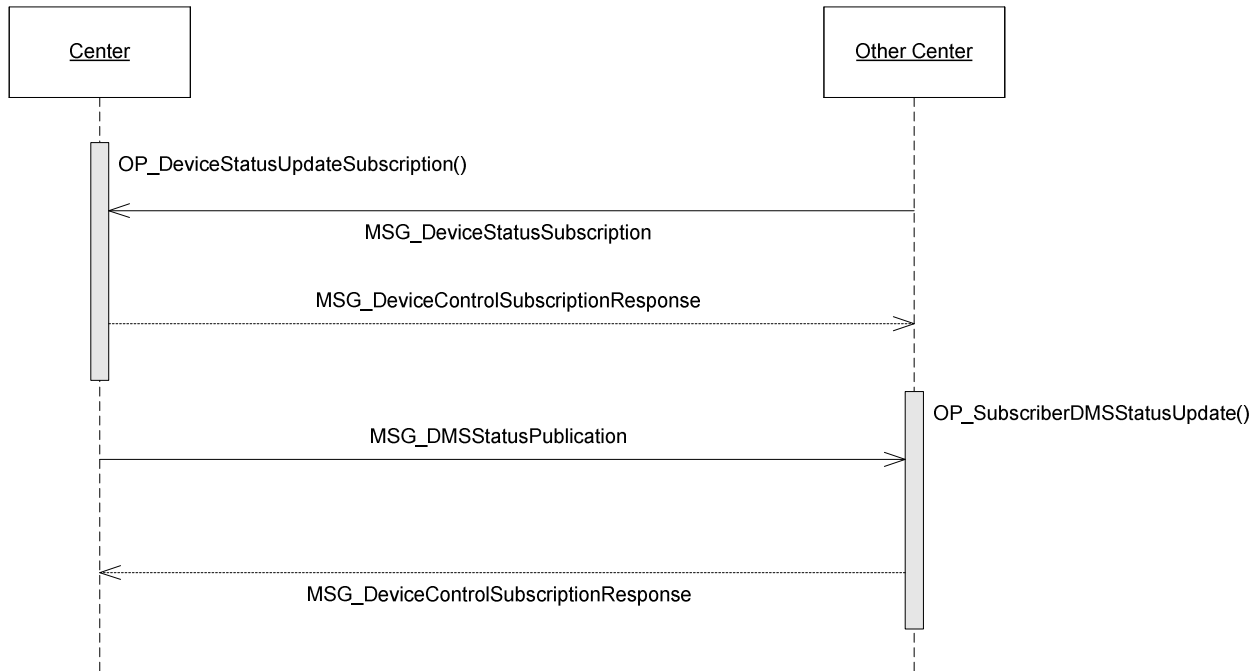




#### 4.1.5 Dialog – DL\_DMSStatusUpdateSubscriptionPublication

The figure below illustrates the operations, message inputs and outputs of the DMSStatusUpdateSubscriptionPublication Dialog.

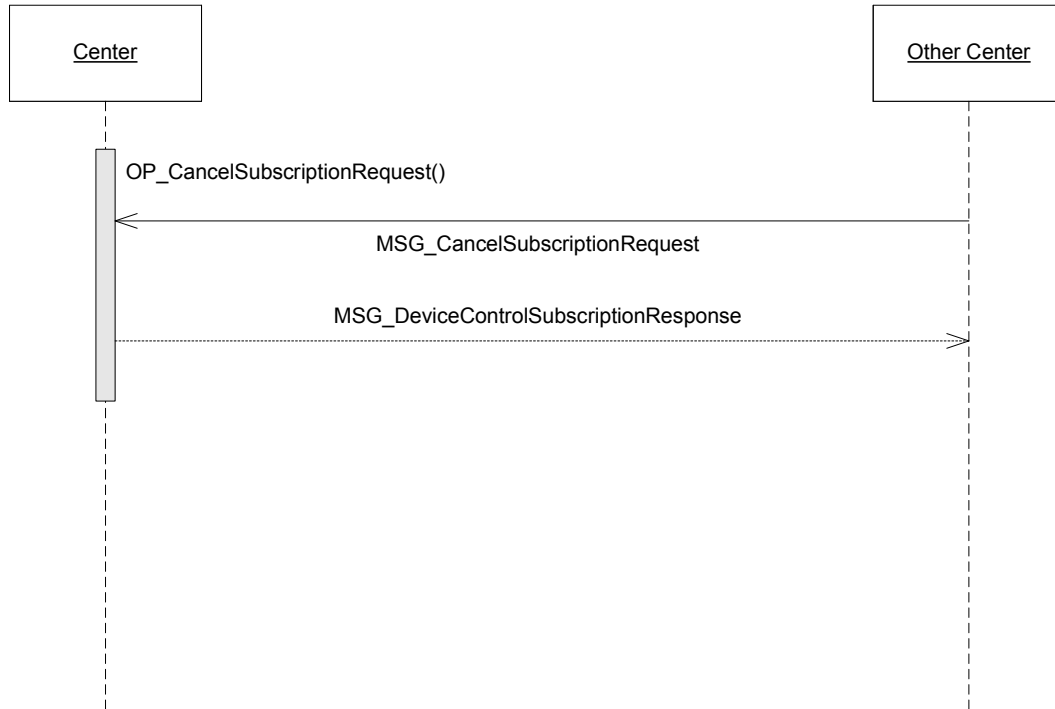
Figure 4-5. DL\_DMSStatusUpdateSubscriptionPublication



#### 4.1.6 Dialog – DL\_CancelSubscriptionRequest

The figure below illustrates the operations, message inputs and outputs of the CancelSubscriptionRequest Dialog.

**Figure 4-6. DL\_CancelSubscriptionRequest**



#### 4.1.7 Dialog – DL\_DeviceCancelControlRequest

The figure below illustrates the operations, message inputs and outputs of the DeviceCancelControlRequest Dialog.

Figure 4-7. DL\_DeviceCancelControlRequest



#### 4.2 Web Services Description Language Worksheet

The worksheet provided below lists the system interface elements (operations, message encoding, message inputs, message outputs, and transport) in table form. The worksheet was developed through analysis of the required dialogs, shown above. This table provides the information necessary to further develop the WSDL.

**Table 4-1. Web Services Definition Language Worksheet**

Service	Encoding	Transport	Operation	Message Input	Message Output	Msg. Pattern
nystateSoapHttpService	SOAP	HTTP	OP_CancelSubscriptionRequest	MSG_DMSPControlRequest	MSG_DeviceControlSubscriptionResponse	R/R
nystateSoapHttpService	SOAP	HTTP	OP_DMSPControlRequest	MSG_DMSPControlRequest	MSG_DeviceControlSubscriptionResponse	R/R
nystateSoapHttpService	SOAP	HTTP	OP_DMSPInventoryRequest	MSG_DeviceInventoryRequest	MSG_DMSPInventoryResponse	R/R
nystateSoapHttpService	SOAP	HTTP	OP_DMSPStatusRequest	MSG_DeviceStatusRequest	MSG_DMSPStatusResponse	R/R
nystateSoapHttpService	SOAP	HTTP	OP_DeviceArchiveStatusSubscription	MSG_DeviceArchiveStatusSubscription	MSG_DeviceControlSubscriptionResponse	Sub
nystateSoapHttpService	SOAP	HTTP	OP_DeviceCancelControlRequest	MSG_DeviceCancelControlRequest	MSG_DeviceControlSubscriptionResponse	R/R
nystateSoapHttpService	SOAP	HTTP	OP_DeviceInventoryUpdateSubscription	MSG_DeviceInventorySubscription	MSG_DeviceControlSubscriptionResponse	Sub
nystateSoapHttpService	SOAP	HTTP	OP_DeviceStatusUpdateSubscription	MSG_DeviceStatusSubscription	MSG_DeviceControlSubscriptionResponse	Sub
nystateSubscriberCallbackService	SOAP	HTTP	OP_SubscriberDMSPInventoryUpdate	MSG_DMSPInventoryPublication	MSG_DeviceControlSubscriptionResponse	Pub
nystateSubscriberCallbackService	SOAP	HTTP	OP_SubscriberDMSPStatusUpdate	MSG_DMSPStatusPublication	MSG_DeviceControlSubscriptionResponse	Pub

**MSG Patterns:**

R/R - Request/Response

Sub – Subscription

Pub – Publication

1-Way - One-Way

**Table 4-2. Mapping of WSDL Message Names to XML Schema Elements**

Message	SchemaElement
MSG_CancelSubscriptionRequest	nystate:cancelSubscriptionRequest
MSG_DeviceArchiveStatusSubscription	nystate:deviceArchiveStatusSubscription
MSG_DeviceCancelControlRequest	nystate:deviceCancelControlRequest
MSG_DeviceControlSubscriptionResponse	nystate:deviceControlSubscriptionResponse
MSG_DeviceInventoryRequest	nystate:deviceInventoryRequest
MSG_DeviceInventorySubscription	nystate:deviceInventorySubscription
MSG_DeviceStatusRequest	nystate:deviceStatusRequest
MSG_DeviceStatusSubscription	nystate:deviceStatusSubscription
MSG_DMSPControlRequest	nystate:dMSPControlRequest
MSG_DMSPInventoryResponse	nystate:dMSPInventoryResponse
MSG_DMSPInventoryPublication	nystate:dMSPInventoryPublication
MSG_DMSPStatusResponse	nystate:dMSPStatusResponse
MSG_DMSPStatusPublication	nystate:dMSPStatusPublication



## 5 Center Interface Message, Data Frame, and Data Element Definitions

This section lists all the data concepts used in the NYState ITS Project XML schema. Note that when a data concept was used, it was taken from another standard and was moved into the nystate namespace to avoid confusion with the ITS standard from which it came. Messages used in this specification have the term "" added to their descriptive, ASN and XML names to aid in this as well. Therefore, any namespace prefixes in the elements have been removed and all elements appear in the common local namespace. This has produced a "flat" schema with no further imports.

Text dealing with data concepts meaning were extracted (and then revised) from the final adopted TMDD V2.1 and produce this document.

The original TMDD V2.1 ASN.1 description has been included in the descriptions below. Data concepts not applicable to the project were commented out and a comment added to indicate new project-specific elements as needed. The XML representation indicates the "new" nystate data concept.

### 5.1 #1 AuthorizationSet (Data Frame)

**Definition:** A data frame which contains the authorization information provided by a user as credentials for a message. Typically this information is validated against an X.509 certificate using XMLSEC methods. The contents of this data frame should be suppressed if the message is to be relayed to other down stream machines after the authorization has completed. The entire message should be deleted if the authorization fails.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
AuthorizationSet ::= SEQUENCE {
    user-id      Security-user-name,
                -- #3108
    password     Security-password,
                -- #3109
    hashtime     DateTimePair,
                -- XML style date and time
    hash         IA5String (SIZE(10..20)),
                -- the securty hash determined by
                -- the x.509 cert in base-64 form
    ...
}
```

#### XML Representation:

```
<xs:complexType name="AuthorizationSet" >
  <xs:sequence>
    <xs:element name="user-id" type="Security-user-name" />
    <xs:element name="password" type="Security-password" />
    <xs:element name="hashtime" type="DateTimePair" />
    <xs:element name="hash" >
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="10"/>
          <xs:maxLength value="20"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
```

```
</xs:complexType>
```

## 5.2 #2 BroadcastAlerts (Data Frame)

**Definition:** A collection of BroadcastAlertsItem items

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
BroadcastAlerts ::= SEQUENCE (SIZE(1..2)) OF BroadcastAlertsItem
-- with only two values we really do not need this
```

**XML Representation:**

```
<xs:complexType name="BroadcastAlerts" >
  <xs:sequence minOccurs="1" maxOccurs="2">
    <xs:element name="broadcastAlert" type="BroadcastAlertsItem" />
  </xs:sequence>
</xs:complexType>
```

## 5.3 #3 BroadcastAlertsItem (Data Element)

**Definition:** An enumerated list of items.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
BroadcastAlertsItem ::= ENUMERATED {
  reserved (0),
  broadcastAlertsAccepted (1),
  broadcastAlertsNotAccepted (2),
  ...
}
```

**XML Representation:**

```
<xs:simpleType name="BroadcastAlertsItem" >
  <xs:annotation>
    <xs:appinfo>
      reserved (0)
      broadcastAlertsAccepted (1)
      broadcastAlertsNotAccepted (2)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="2"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="broadcastAlertsAccepted"/>
        <xs:enumeration value="broadcastAlertsNotAccepted"/>
      </xs:restriction>
    </xs:simpleType >
  </xs:union>
</xs:simpleType>
```

## 5.4 #4 C2C Message Publication (Data Frame)

**Definition:** This is a data frame used in the response to the subscription process.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
C2cMessagePublication ::= SEQUENCE {
  informationalText InformationalText OPTIONAL,
```

subscriptionID	SubscriptionID,
subscriptionName	SubscriptionName OPTIONAL,
subscriptionFreq	SubscriptionFrequency OPTIONAL,
subscriptionCount	SubscriptionCount OPTIONAL,
...	

### XML Representation:

```
<xs:complexType name="C2cMessagePublication" >
  <xs:sequence>
    <xs:element name="informationalText" type="InformationalText" minOccurs="0"/>
    <xs:element name="subscriptionID" type="SubscriptionID" />
    <xs:element name="subscriptionName" type="SubscriptionName" minOccurs="0"/>
    <xs:element name="subscriptionFreq" type="SubscriptionFrequency" minOccurs="0"/>
    <xs:element name="subscriptionCount" type="SubscriptionCount" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

## 5.5 #5 C2C Message Subscription (Data Frame)

**Definition:** This is a data frame used in the subscription process.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
C2cMessageSubscription ::= SEQUENCE {
  informationalText      InformationalText OPTIONAL,
  returnAddress          ReturnAddress,
  subscriptionAction     SubscriptionAction,
  subscriptionType       SubscriptionType,
  subscriptionID         SubscriptionID,
  subscriptionName       SubscriptionName OPTIONAL,
  subscriptionTimeFrame  SubscriptionTimeFrame OPTIONAL,
  subscriptionFrequency  SubscriptionFrequency,
  broadcastAlerts        BroadcastAlerts OPTIONAL,
  ...
}
```

### XML Representation:

```
<xs:complexType name="C2cMessageSubscription" >
  <xs:sequence>
    <xs:element name="informationalText" type="InformationalText" minOccurs="0"/>
    <xs:element name="returnAddress" type="ReturnAddress" />
    <xs:element name="subscriptionAction" type="SubscriptionAction" />
    <xs:element name="subscriptionType" type="SubscriptionType" />
    <xs:element name="subscriptionID" type="SubscriptionID" />
    <xs:element name="subscriptionName" type="SubscriptionName" minOccurs="0"/>
    <xs:element name="subscriptionTimeFrame" type="SubscriptionTimeFrame" minOccurs="0"/>
    <xs:element name="subscriptionFrequency" type="SubscriptionFrequency" />
    <xs:element name="broadcastAlerts" type="BroadcastAlerts" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

## 5.6 #6 C2C\_MessageReceipt (Data Frame)

**Definition:** AAA An empty definition field.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
C2cMessageReceipt ::= SEQUENCE {
  informationalText      InformationalText,
  ...
}
```

### XML Representation:

```
<xs:complexType name="C2cMessageReceipt" >
  <xs:sequence>
    <xs:element name="informationalText" type="InformationalText" />
  </xs:sequence>
</xs:complexType>
```



```

        </xs:sequence>
    </xs:complexType>

```

## 5.7 #7 C2C\_SubscriptionTimeFrame (Data Frame)

**Definition:** Time frame during which (the) subscriber requests that (the) publication be active.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```

SubscriptionTimeFrame ::= SEQUENCE {
    start    DateTimePair,
    end      DateTimePair,
    ...
} -- using XML values

```

**XML Representation:**

```

<xs:complexType name="SubscriptionTimeFrame" >
    <xs:sequence>
        <xs:element name="start" type="DateTimePair" />
        <xs:element name="end" type="DateTimePair" />
    </xs:sequence>
</xs:complexType>

```

## 5.8 #8 CancelSubscriptionRequest (Message)

**Definition:** C2C - Cancel Subscription Request Message, Based on Device Cancel Control Request Messages for DMS and Signals.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```

CancelSubscriptionRequest ::= SEQUENCE {
    organization-owning      OrganizationInformationShort,
    organization-requesting  OrganizationInformationLong,
    authorization            AuthorizationSet,
    request-id               Device-request-identifier, -- #3108
    ...
}

```

**XML Representation:**

```

<xs:element name="cancelSubscriptionRequest" type="CancelSubscriptionRequest"/>
<xs:complexType name="CancelSubscriptionRequest" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>

```

## 5.9 #9 CONTACT\_EmailAddress\_text (Data Element)

**Definition:** The e-mail address of a person at an organization to contact regarding a roadway event.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-email-address ::= IA5String (SIZE(1..128)) -- Any set of ASCII characters up to 128

**XML Representation:**

```

<xs:simpleType name="Contact-email-address" >
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="128"/>
    </xs:restriction>
</xs:simpleType>

```

**5.10 #10 CONTACT\_Identifier\_identifier (Data Element)**

**Definition:** A unique identifier for a contact person as assigned by the organization.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-identifier ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

**XML Representation:**

```
<xs:simpleType name="Contact-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

**5.11 #11 CONTACT\_PersonName\_text (Data Element)**

**Definition:** The name or identifier of a person at an organization.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-person-name ::= IA5String (SIZE(1..64)) -- Any set of alphanumeric characters up to 64

**XML Representation:**

```
<xs:simpleType name="Contact-person-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>
```

**5.12 #12 CONTACT\_PersonTitle\_text (Data Element)**

**Definition:** The title of a person at an organization.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-person-title ::= IA5String (SIZE(1..64)) -- Any set of alphanumeric characters up to 64

**XML Representation:**

```
<xs:simpleType name="Contact-person-title" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>
```

**5.13 #13 CONTACT\_PhoneAlternate\_text (Data Element)**

**Definition:** An alternate phone number (e.g., home phone number) of a person at an organization.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-phone-alternate ::= IA5String (SIZE(1..32)) -- Any set of ASCII characters up to 32

**XML Representation:**

```
<xs:simpleType name="Contact-phone-alternate" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

**5.14 #14 CONTACT\_PhoneNumber\_text (Data Element)**

**Definition:** The telephone number of a person at an organization.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Contact-phone-number ::= IA5String (SIZE(1..32)) -- Any set of ASCII characters up to 32. The '-' and '.' characters shall be valid delimiters.

**XML Representation:**

```
<xs:simpleType name="Contact-phone-number" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

**5.15 #15 ContactDetails (Data Frame)**

**Definition:** A collection of contact data elements defined by TMDD and refined for use by DOT

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
ContactDetails ::= SEQUENCE
{
  contact-id          Contact-identifier,          --3105
  person-name         Contact-person-name OPTIONAL, --3206
  person-title        Contact-person-title OPTIONAL, --3349
  -- organization-id   Organization-identifier OPTIONAL, --#-3343
  -- organization-name Organization-name OPTIONAL,      --#-3344
  phone-number        Contact-phone-number OPTIONAL,   --3207
  phone-alternate     Contact-phone-alternate OPTIONAL, --3113
  -- mobile-number    Contact-mobile-phone-number OPTIONAL, --#-3350
  -- fax-number       Contact-phone-fax OPTIONAL,        --#-3205
  -- pager-id        Contact-pager-identifier OPTIONAL,  --#-3346
  -- pager-number    Contact-pager-number OPTIONAL,      --#-3347
  email-address       Contact-email-address OPTIONAL,    --3204
  -- radio-unit      Contact-radio-unit-identifier OPTIONAL, --#-3208
  -- address-line1   Contact-mailing-address-line1 OPTIONAL, --#-3339
  -- address-line2   Contact-mailing-address-line2 OPTIONAL, --#-3340
  -- city            Contact-mailing-address-city OPTIONAL,  --#-3338
  -- state           Contact-mailing-address-state OPTIONAL, --#-3341
  -- zip-code        Contact-mailing-address-zip OPTIONAL,   --#-3342
  -- country         Contact-mailing-address-country OPTIONAL --#-3373
  ...
}
```

**XML Representation:**

```
<xs:complexType name="ContactDetails" >
  <xs:sequence>
    <xs:element name="contact-id" type="Contact-identifier" />
    <xs:element name="person-name" type="Contact-person-name" minOccurs="0"/>
    <xs:element name="person-title" type="Contact-person-title" minOccurs="0"/>
    <xs:element name="phone-number" type="Contact-phone-number" minOccurs="0"/>
    <xs:element name="phone-alternate" type="Contact-phone-alternate" minOccurs="0"/>
    <xs:element name="email-address" type="Contact-email-address" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

**5.16 #16 DateTimePair (Data Frame)**

**Definition:** A representation of date, time, and time zone in the standard XML formats.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
DateTimePair ::= SEQUENCE {
```

```

date      Date,
time      Time,
offset    Time,
...
} -- expressed in basic XML date and time

```

### XML Representation:

```

<xs:simpleType name="DateTimePair">
  <xs:restriction base="xs:dateTime"/>
</xs:simpleType>
<!-- Native XML formats are used for date and time -->

```

## 5.17 #17 DEVICE\_AcknowledgeControl\_code (Data Element)

**Definition:** Acknowledgement of request command from one TMC to another for device action.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```

Device-acknowledge-control ::= ENUMERATED {
  control-acknowledged      (0),
  device-available          (1),
  requested-changes-completed (2),
  control-rejected          (3),
  device-in-use             (4),
  device-off-line           (5),
  request-cancelled         (6),
  request-rejected-invalid-command-parameters (7),
  request-rejected-insufficient-privileges-of-the-requesting-operator (8),
  request-queued            (9),
  ...
}

```

### XML Representation:

```

<xs:simpleType name="Device-acknowledge-control" >
  <xs:annotation>
    <xs:appinfo>
      control acknowledged (0)
      device available (1)
      requested changes completed (2)
      control rejected (3)
      device in use (4)
      device off line (5)
      request cancelled (6)
      request rejected invalid command parameters (7)
      request rejected insufficient privileges of the requesting operator (8)
      request queued (9)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="9"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="control acknowledged"/>
        <xs:enumeration value="device available"/>
        <xs:enumeration value="requested changes completed"/>
        <xs:enumeration value="control rejected"/>
        <xs:enumeration value="device in use"/>
        <xs:enumeration value="device off line"/>
        <xs:enumeration value="request cancelled"/>
        <xs:enumeration value="request rejected invalid command parameters"/>
        <xs:enumeration value="request rejected insufficient privileges of the requesting operator"/>
        <xs:enumeration value="request queued"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

```

        </xs:simpleType >
    </xs:union>
</xs:simpleType>

```

### 5.18 #18 *DEVICE\_CommandEndTime\_utc (Data Element)*

**Definition:** Expiration time of the command requested for the device control

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Device-command-end-time ::= INTEGER -- represented as HHMMSS where HH is the hour of the day in 24-hour clock form (with legal values between 00 and 23); "MM" is the minute of hour (with legal values from 00 to 59); "SS" is the second of the minute (with legal values from 00 to 59).

**XML Representation:**

```

<xs:simpleType name="Device-command-end-time" >
    <xs:restriction base="xs:int"/>
</xs:simpleType>

```

### 5.19 #19 *DEVICE\_CommandRequestPriority\_number (Data Element)*

**Definition:** A code to indicate universal concept of priority for all command requests. If the request is accepted and the action is posted, then the current state has that priority.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Device-command-request-priority ::= INTEGER (1..10) -- Select from 1 to 10 where 1 is the highest and 10 is the lowest priority

**XML Representation:**

```

<xs:simpleType name="Device-command-request-priority" >
    <xs:restriction base="xs:unsignedByte">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="10"/>
    </xs:restriction>
</xs:simpleType>

```

### 5.20 #20 *DEVICE\_Identifier\_identifier (Data Element)*

**Definition:** A unique alphanumeric device identifier.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Device-identifier ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

**XML Representation:**

```

<xs:simpleType name="Device-identifier" >
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="32"/>
    </xs:restriction>
</xs:simpleType>

```

### 5.21 #21 *DEVICE\_OperationalStatus\_code (Data Element)*

**Definition:** Operational status of a device (e.g., on, off, etc.).

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```

Device-operational-status ::= ENUMERATED {
    on          (1),
    off         (2),

```

```

in-service      (3),
out-of-service  (4),
unavailable     (5),
unknown        (6),
...
}

```

### XML Representation:

```

<xs:simpleType name="Device-operational-status" >
  <xs:annotation>
    <xs:appinfo>
      on (1)
      off (2)
      in service (3)
      out of service (4)
      unavailable (5)
      unknown (6)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="6"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="on"/>
        <xs:enumeration value="off"/>
        <xs:enumeration value="in service"/>
        <xs:enumeration value="out of service"/>
        <xs:enumeration value="unavailable"/>
        <xs:enumeration value="unknown"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

## 5.22 #22 DEVICE\_RequestIdentifier\_identifier (Data Element)

**Definition:** A unique sequence number generated by the requesting center that uniquely identifies the control request within the requesting center.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Device-request-identifier ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

### XML Representation:

```

<xs:simpleType name="Device-request-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

```

## 5.23 #23 DEVICE\_Type\_code (Data Element)

**Definition:** A code which specifies the type of device.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```

Device-type ::= ENUMERATED {
  detector          (1),
  cctv-camera      (2),

```

```
dynamic-message-sign      (3),
environmental-sensor-station (4),
gate                     (5),
highway-advisory-radio    (6),
lane-control-signal       (7),
ramp-meter               (8),
signal-controller         (9),
signal-section            (10),
...
}
```

### XML Representation:

```
<xs:simpleType name="Device-type" >
  <xs:annotation>
    <xs:appinfo>
      detector (1)
      cctv camera (2)
      dynamic message sign (3)
      environmental sensor station (4)
      gate (5)
      highway advisory radio (6)
      lane control signal (7)
      ramp meter (8)
      signal controller (9)
      signal section (10)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="10"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="detector"/>
        <xs:enumeration value="cctv camera"/>
        <xs:enumeration value="dynamic message sign"/>
        <xs:enumeration value="environmental sensor station"/>
        <xs:enumeration value="gate"/>
        <xs:enumeration value="highway advisory radio"/>
        <xs:enumeration value="lane control signal"/>
        <xs:enumeration value="ramp meter"/>
        <xs:enumeration value="signal controller"/>
        <xs:enumeration value="signal section"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>
```

## 5.24 #24 DeviceArchiveStatusSubscription (Message)

**Definition:** C2C - Device Archive Status Information Subscription

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceArchiveStatusSubscription ::= SEQUENCE {
  organization-owning      OrganizationInformationShort,
  organization-requesting  OrganizationInformationLong,
  authorization            AuthorizationSet,
  request-id              Device-request-identifier, -- #3108
  command-start-time      DateTimePair,
  command-end-time        DateTimePair,
  -- note that XML time is used

  -- Items below added by DOT to meet local needs
  archive-frequency        TimeInterval,
  request-send-time        Time,
```

```
request-send-date      Date,
                      -- consider combing to make
                      -- one time element here

filters               Filters,
...
}
```

### XML Representation:

```
<xs:element name="deviceArchiveStatusSubscription" type="DeviceArchiveStatusSubscription"/>
<xs:complexType name="DeviceArchiveStatusSubscription" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="request-id" type="Device-request-identifier" />
    <xs:element name="command-start-time" type="DateTimePair" />
    <xs:element name="command-end-time" type="DateTimePair" />
    <xs:element name="archive-frequency" type="TimeInterval" />
    <xs:element name="request-send-time" type="Time" />
    <xs:element name="request-send-date" type="Date" />
    <xs:element name="filters" type="Filters" />
  </xs:sequence>
</xs:complexType>
```

## 5.25 #25 DeviceCancelControlRequest (Message)

**Definition:** C2C - Device Cancel Control Request Message, Based on TMDD Device Control Response Messages for DMS and Signals

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceCancelControlRequest ::= SEQUENCE {
  organization-owning      OrganizationInformationShort,
  organization-requesting  OrganizationInformationLong,
  authorization             AuthorizationSet,

  device-type              Device-type, -- #3747
  device-id                Device-identifier, --#3701
  request-id               Device-request-identifier, --#3717

  -- Items below added by DOT to meet local needs
  freeText                 FreeText,
  ...
}
```

### XML Representation:

```
<xs:element name="deviceCancelControlRequest" type="DeviceCancelControlRequest"/>
<xs:complexType name="DeviceCancelControlRequest" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="device-type" type="Device-type" />
    <xs:element name="device-id" type="Device-identifier" />
    <xs:element name="request-id" type="Device-request-identifier" />
    <xs:element name="freeText" type="FreeText" />
  </xs:sequence>
</xs:complexType>
```

## 5.26 #26 DeviceControlSubscriptionResponse (Message)

**Definition:** C2C - Device Message Response

Based on TMDD **Device Control Response** Messages for DMS and Signals Also used to acknowledge the following: a Device Inventory Update Subscriptions for DMS and Signals, Device Inventory Update



Subscriptions, Device Status Update Subscriptions, Subscription Cancellations, Archival Subscription Cancellations.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceControlSubscriptionResponse ::= SEQUENCE
{
    organization-owning      OrganizationInformationLong,
    organization-requesting  OrganizationInformationShort,
    authorization            AuthorizationSet,

    request-id              Device-request-identifier, --#3717
    device-id               Device-identifier, --#3701
    request-status          Device-acknowledge-control,

    -- Below in std but not used
    -- center-id            Organization-center-identifier, --#3217
    -- operator-id          Device-organization-operator-identifier, --#3706
    -- request-response     Device-acknowledge-control, --#3763
    -- operator-last-revised Event-update-operator-last-revised OPTIONAL, --#3295
    -- response-plan-id     Event-response-plan-identifier OPTIONAL --#3269

    -- Items below added by DOT to meet local needs
    freeText                FreeText,
    ...
}
```

### XML Representation:

```
<xs:element name="deviceControlSubscriptionResponse" type="DeviceControlSubscriptionResponse" />
<xs:complexType name="DeviceControlSubscriptionResponse" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationLong" />
        <xs:element name="organization-requesting" type="OrganizationInformationShort" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="request-id" type="Device-request-identifier" />
        <xs:element name="device-id" type="Device-identifier" />
        <xs:element name="request-status" type="Device-acknowledge-control" />
        <xs:element name="freeText" type="FreeText" />
    </xs:sequence>
</xs:complexType>
```

## 5.27 #27 DeviceInventoryRequest (Message)

**Definition:** C2C - Device Inventory Request and Subscription Message, Based on TMDD DMS and Intersection (Signal) Inventory Request Messages

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceInventoryRequest ::= SEQUENCE {
    organization-owning      OrganizationInformationShort,
    organization-requesting  OrganizationInformationLong,
    authorization            AuthorizationSet,

    device-type              Device-type, -- 3747
    -- does having Device-type here limit us to requesting
    -- only one type of device in the next line??

    device-list SEQUENCE (SIZE(1..100000)) OF
        Device-identifier, -- 3701
        -- an id = 0 means request all

    -- Items below added by DOT to meet local needs
    request-id              Device-request-identifier, -- 3717
    ...
}
```

### XML Representation:

```
<xs:element name="deviceInventoryRequest" type="DeviceInventoryRequest"/>
<xs:complexType name="DeviceInventoryRequest" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="device-type" type="Device-type" />
    <xs:element name="device-list" >
      <xs:complexType>
        <xs:sequence minOccurs="1" maxOccurs="100000">
          <xs:element name="device" type="Device-identifier" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="request-id" type="Device-request-identifier" />
  </xs:sequence>
</xs:complexType>
```

### 5.28 #28 DeviceInventorySubscription (Message)

**Definition:** C2C - Device Inventory Subscription Message. Based on Device Inventory Request Message  
Content of message enclosed in SOAP Body

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceInventorySubscription ::= SEQUENCE {
  c2cMessageSubscription C2cMessageSubscription,
  deviceInventoryRequest DeviceInventoryRequest,
  ...
}
```

#### XML Representation:

```
<xs:element name="deviceInventorySubscription" type="DeviceInventorySubscription"/>
<xs:complexType name="DeviceInventorySubscription" >
  <xs:sequence>
    <xs:element name="c2cMessageSubscription" type="C2cMessageSubscription" />
    <xs:element name="deviceInventoryRequest" type="DeviceInventoryRequest" />
  </xs:sequence>
</xs:complexType>
```

### 5.29 #29 DeviceStatusRequest (Message)

**Definition:** C2C - Device Status Information Request Message

Based on TMDD DMS and Intersection (Signal) Status Request Messages

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DeviceStatusRequest ::= SEQUENCE {
  organization-owning OrganizationInformationShort,
  organization-requesting OrganizationInformationLong,
  authorization AuthorizationSet,

  -- same question for device type here as before
  device-type Device-type, -- 3747
  device-list SEQUENCE (SIZE(1..100000)) OF
    Device-identifier, -- 3701
    -- an id = 0 means request all

  -- Items below added by DOT to meet local needs
  request-id Device-request-identifier, -- 3717
  ...
}
```

#### XML Representation:

```
<xs:element name="deviceStatusRequest" type="DeviceStatusRequest"/>
```

```
<xs:complexType name="DeviceStatusRequest" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="device-type" type="Device-type" />
    <xs:element name="device-list" >
      <xs:complexType>
        <xs:sequence minOccurs="1" maxOccurs="100000">
          <xs:element name="device" type="Device-identifier" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="request-id" type="Device-request-identifier" />
  </xs:sequence>
</xs:complexType>
```

### **5.30 #30 DeviceStatusSubscription (Message)**

**Definition:** C2C - Device Status Subscription Message, Based on Device Status Request Message  
Content of message enclosed in SOAP Body

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
DeviceStatusSubscription ::= SEQUENCE {
  c2cMessageSubscription C2cMessageSubscription,
  deviceStatusRequest    DeviceStatusRequest,
  ...
}
```

**XML Representation:**

```
<xs:element name="deviceStatusSubscription" type="DeviceStatusSubscription"/>
<xs:complexType name="DeviceStatusSubscription" >
  <xs:sequence>
    <xs:element name="c2cMessageSubscription" type="C2cMessageSubscription" />
    <xs:element name="deviceStatusRequest" type="DeviceStatusRequest" />
  </xs:sequence>
</xs:complexType>
```

### **5.31 #31 DMSControlRequest (Message)**

**Definition:** C2C - DMS Control Request Message, Based on TMDD DMS Control Request Message

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
DMSControlRequest ::= SEQUENCE {
  organization-owning      OrganizationInformationShort,
  organization-requesting  OrganizationInformationLong,
  authorization            AuthorizationSet,

  device-id               Device-identifier, -- 3701
  request-id              Device-request-identifier, -- 3717
  dms-beacon-control      DmsMessageMultiString, -- 3900
  dms-message             DmsMessageMultiString, -- 3901
  message-number          DmsMessageNumber, -- 3902
  command-request-priority Device-command-request-priority, -- 3711
  command-start-time      DateTimePair,
  command-end-time        DateTimePair,

  -- Items below added by DOT to meet local needs
  freeText                FreeText,
  ...
}
```

**XML Representation:**

```
<xs:element name="dMSControlRequest" type="DMSControlRequest"/>
```

```
<xs:complexType name="DMSControlRequest" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="device-id" type="Device-identifier" />
    <xs:element name="request-id" type="Device-request-identifier" />
    <xs:element name="dms-beacon-control" type="DmsMessageMultiString" />
    <xs:element name="dms-message" type="DmsMessageMultiString" />
    <xs:element name="message-number" type="DmsMessageNumber" />
    <xs:element name="command-request-priority" type="Device-command-request-priority" />
    <xs:element name="command-start-time" type="DateTimePair" />
    <xs:element name="command-end-time" type="DateTimePair" />
    <xs:element name="freeText" type="FreeText" />
  </xs:sequence>
</xs:complexType>
```

### 5.32 #32 DMSDeviceStatus (Data Frame)

**Definition:** In TMDD this is documented a message, here is it used as a data frame.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DMSDeviceStatus ::= SEQUENCE
{
  -- organization-information OrganizationInformation,
  -- Confirm above is never in fact used or needed
  -- operator-id Organization-center-operator-identifier, --3112
  device-id Device-identifier, --3701
  dms-device-status Device-operational-status, --3716
  dms-current-message DmsMessageMultiString, --3901
  message-time-remaining DmsMessageTimeRemaining OPTIONAL, --3904
  -- message-source-mode NTCIP.DmsMsgSourceMode OPTIONAL, --3905
  -- associated-event-id Event-response-plan-identifier OPTIONAL, --3269
  last-comm-time DateTimePair,
  ...
}
```

#### XML Representation:

```
<xs:element name="dmsDeviceStatus" type="DMSDeviceStatus"/>
<xs:complexType name="DMSDeviceStatus" >
  <xs:sequence>
    <xs:element name="device-id" type="Device-identifier" />
    <xs:element name="dms-device-status" type="Device-operational-status" />
    <xs:element name="dms-current-message" type="DmsMessageMultiString" />
    <xs:element name="message-time-remaining" type="DmsMessageTimeRemaining"
minOccurs="0"/>
    <xs:element name="last-comm-time" type="DateTimePair" />
  </xs:sequence>
</xs:complexType>
```

### 5.33 #33 DMSInventoryPublication (Message)

**Definition:** C2C - DMS Inventory Update Publication Message

Based on C2C DMS Inventory Response Message

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DMSInventoryPublication ::= SEQUENCE {
  c2cMessagePublication C2cMessagePublication,
  dmsInventory DMSInventoryResponse,
  ...
}
```

#### XML Representation:

```
<xs:element name="dmsInventoryPublication" type="DMSInventoryPublication"/>
```

```
<xs:complexType name="DMSInventoryPublication" >
  <xs:sequence>
    <xs:element name="c2cMessagePublication" type="C2cMessagePublication" />
    <xs:element name="dMSInventory" type="DMSInventoryResponse" />
  </xs:sequence>
</xs:complexType>
```

### 5.34 #34 DMSInventoryResponse (Message)

**Definition:** C2C - DMS Inventory Response Message, Based on TMDD DMS Inventory Message and the DMSDeviceStatus data frame.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DMSInventoryResponse ::= SEQUENCE {
  organization-owning      OrganizationInformationLong,
  organization-requesting  OrganizationInformationShort,

  devices SEQUENCE (SIZE(0..10000)) OF
    DMSDeviceStatus,
    -- note DOT additions here

  -- Items below added by DOT to meet local needs
  request-id              Device-request-identifier, -- 3717
  ...
}
```

#### XML Representation:

```
<xs:element name="dMSInventoryResponse" type="DMSInventoryResponse"/>
<xs:complexType name="DMSInventoryResponse" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationLong" />
    <xs:element name="organization-requesting" type="OrganizationInformationShort" />
    <xs:element name="devices" >
      <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs="10000">
          <xs:element name="device" type="DMSDeviceStatus" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="request-id" type="Device-request-identifier" />
  </xs:sequence>
</xs:complexType>
```

### 5.35 #35 DMSStatusPublication (Message)

**Definition:** C2C - DMS Status Information Response Message used in replying to a subscription request.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DMSStatusPublication ::= SEQUENCE {
  c2cMessagePublication  C2cMessagePublication,
  dMSStatus              DMSStatusResponse,
  ...
}
```

#### XML Representation:

```
<xs:element name="dMSStatusPublication" type="DMSStatusPublication"/>
<xs:complexType name="DMSStatusPublication" >
  <xs:sequence>
    <xs:element name="c2cMessagePublication" type="C2cMessagePublication" />
    <xs:element name="dMSStatus" type="DMSStatusResponse" />
  </xs:sequence>
</xs:complexType>
```

### 5.36 #36 *DMSStatusResponse* (Message)

**Definition:** C2C - DMS Status Information Response Message

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
DMSStatusResponse ::= SEQUENCE {
    organization-owning      OrganizationInformationShort,
    organization-requesting  OrganizationInformationLong,
    authorization            AuthorizationSet,

    devices SEQUENCE (SIZE(0..10000)) OF
        DMSDeviceStatus,
        -- using: TMDD.DMSDeviceStatus
        -- note DOT additions here

    -- Items below added by DOT to meet local needs
    request-id              Device-request-identifier, -- 3717
    ...
}
```

**XML Representation:**

```
<xs:element name="dMSStatusResponse" type="DMSStatusResponse"/>
<xs:complexType name="DMSStatusResponse" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="devices" >
            <xs:complexType>
                <xs:sequence minOccurs="0" maxOccurs="10000">
                    <xs:element name="device" type="DMSDeviceStatus" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>
```

### 5.37 #37 *EXT\_ATIS\_Date* (Data Element)

**Definition:** Date for which a directory entry, or other use, is being requested by a Traveler or used in a returned message. The precise time at which this date starts may vary with the time zone of the service. For example: Saturday stay over rates for an airline begin at the point of departure.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
Date ::= IA5String (SIZE(8))
    -- format as: YYYYMMDD where
    -- YYYY the year, in common era units
    -- MM the month, range 01 to 12
    -- DD the day, range 01 to 31
    -- use zero for MM and DD when not applicable
    -- The entry: xs:date is used for XML
```

**XML Representation:**

```
<xs:simpleType name="Date">
    <xs:restriction base="xs:date"/>
</xs:simpleType>
```

### 5.38 #38 *EXT\_ATIS\_Time* (Data Element)

**Definition:** Time for which a directory entry is being requested by a Traveler or used in a returned message.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
Time ::= IA5String (SIZE(6..10))
-- Format: HHMMSSsss
-- Valid times using 24 hour notation.
-- HH=00 through 23; MM=00 through 59;
-- SS=00 through 59; 00 if NA
-- ssss=0000 through 9999.
-- Use SS= 60 for leap seconds
-- HH represents hours, MM minutes, SS seconds,
-- and ssss decimal seconds to whatever number
-- of significant digits is required (up to four)
-- The entry: xs:time is used for XML
```

### XML Representation:

```
<xs:simpleType name="Time">
  <xs:restriction base="xs:time"/>
</xs:simpleType>
```

## 5.39 #39 EXT\_DmsMessageBeacon (Data Element)

**Definition:** Indicates if connected beacon(s) are to be activated when the associated message is displayed.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DmsMessageBeacon ::= ENUMERATED {
    beacons-are-disabled (0),
    beacons-are-enabled (1),
    ...
}
```

### XML Representation:

```
<xs:simpleType name="DmsMessageBeacon" >
  <xs:annotation>
    <xs:appinfo>
      beacons are disabled (0)
      beacons are enabled (1)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="1"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="beacons are disabled"/>
        <xs:enumeration value="beacons are enabled"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>
```

## 5.40 #40 EXT\_DmsMessageMultiString (Data Element)

**Definition:** Contains the message written in MULTI-language. The value of the MULTI string is not allowed to have any null character.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DmsMessageMultiString ::= OCTET STRING (SIZE(1..256)) -- Any set of ASCII characters up to 256
```

### XML Representation:

```
<xs:complexType name="DmsMessageMultiString" >
```

```
<xs:simpleContent>
  <xs:extension base="DmsMessageMultiString-string" >
    <xs:attribute name="EncodingType" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:NMTOKEN">
          <xs:enumeration value="base64Binary"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:extension>
</xs:simpleContent>
</xs:complexType>
<xs:simpleType name="DmsMessageMultiString-string">
  <xs:restriction base="xs:base64Binary">
    <xs:minLength value="2"/>
    <xs:maxLength value="342"/>
  </xs:restriction>
</xs:simpleType>
```

#### 5.41 #41 EXT\_DmsMessageNumber (Data Element)

**Definition:** Enumerated listing of row entries within the value of the primary index to this table (dmsMessageMemoryType -object). When the primary index is 'currentBuffer' or 'schedule', then this value must be one (1). When the primary index is 'blank', this value shall be from 1 through 255 and all compliant devices must support all 255 of these 'blank' rows.

##### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

DmsMessageNumber ::= INTEGER (0..65535)

##### XML Representation:

```
<xs:simpleType name="DmsMessageNumber" >
  <xs:restriction base="xs:unsignedShort"/>
</xs:simpleType>
```

#### 5.42 #42 EXT\_DmsMessageTimeRemaining (Data Element)

**Definition:** Indicates the amount of remaining time in minutes that the current message shall be displayed. The value 65535 indicates an infinite duration. A value of zero (0) shall indicate that the current message display duration has expired.

##### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

DmsMessageTimeRemaining ::= INTEGER (0..65535)

##### XML Representation:

```
<xs:simpleType name="DmsMessageTimeRemaining" >
  <xs:restriction base="xs:unsignedShort"/>
</xs:simpleType>
```

#### 5.43 #43 EXT\_DmsSignTechnology (Data Element)

**Definition:** Indicates the utilized technology in a bitmap format (Hybrids will have to set the bits for all technologies that the sign utilizes). If a bit is set to one (1), then the associated feature exists.; if the bit is set to zero (0), then the associated feature does not exist.

##### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
DmsSignTechnology ::= BIT STRING {
  other          (0),
```



```

led          (1),
flip-disk    (2),
fiber-optics (3),
shuttered    (4),
bulb         (5),
drum         (6),
spare7       (7) -- Adding to make a full byte
}

```

### XML Representation:

```

<xs:simpleType name="DmsSignTechnology-item" >
  <xs:annotation>
    <xs:appinfo>
      other (0)
      led (1)
      flip disk (2)
      fiber optics (3)
      shuttered (4)
      bulb (5)
      drum (6)
      spare7 (7) -- Adding to make a full byte
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:int">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="7"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="other"/>
        <xs:enumeration value="led"/>
        <xs:enumeration value="flip disk"/>
        <xs:enumeration value="fiber optics"/>
        <xs:enumeration value="shuttered"/>
        <xs:enumeration value="bulb"/>
        <xs:enumeration value="drum"/>
        <xs:enumeration value="spare7"/>
      </xs:restriction>
    </xs:simpleType >
  </xs:union>
</xs:simpleType>
<xs:simpleType name="DmsSignTechnology">
  <xs:list itemType="DmsSignTechnology-item"/>
</xs:simpleType>

```

## 5.44 #44 EXT\_LRMS\_Distance (Data Frame)

**Definition:** A choice of distance values, in integer or REAL expressions, in different English and metric units. Note that the data elements for REAL representations are suffixed with Dec.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```

Distance ::= CHOICE
{
  m      INTEGER (-10000000..10000000),
  mDec   REAL (-10000000..10000000),
  mm     INTEGER (-1000000..1000000),
  mmDec  REAL (-1000000..1000000),
  dm     INTEGER (-1000000..1000000),
  dmDec  REAL (-1000000..1000000),
  yd     INTEGER (-17600000..17600000),
  ydDec  REAL (-17600000..17600000),
  ft     INTEGER (-52800000..52800000),
  ftDec  REAL (-52800000..52800000),
  in     INTEGER (-1000000..1000000),

```

```
inDec REAL (-1000000..1000000),
mi INTEGER (-10000..10000),
miDec REAL (-10000..10000),
km INTEGER (-10000..10000),
kmDec REAL (-10000..10000)
```

```
}
```

## XML Representation:

```
<xs:complexType name="Distance" >
  <xs:choice>
    <xs:element name="m" >
      <xs:simpleType>
        <xs:restriction base="xs:int">
          <xs:minInclusive value="-10000000"/>
          <xs:maxInclusive value="10000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="mDec" >
      <xs:simpleType>
        <xs:restriction base="xs:float">
          <xs:minInclusive value="-10000000"/>
          <xs:maxInclusive value="10000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <!-- Warning, above may need hand editing -->
    <!-- observe that min-max restrictions have -->
    <!-- not been added to this template yet -->
    <xs:element name="mm" >
      <xs:simpleType>
        <xs:restriction base="xs:int">
          <xs:minInclusive value="-1000000"/>
          <xs:maxInclusive value="1000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="mmDec" >
      <xs:simpleType>
        <xs:restriction base="xs:float">
          <xs:minInclusive value="-1000000"/>
          <xs:maxInclusive value="1000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <!-- Warning, above may need hand editing -->
    <!-- observe that min-max restrictions have -->
    <!-- not been added to this template yet -->
    <xs:element name="dm" >
      <xs:simpleType>
        <xs:restriction base="xs:int">
          <xs:minInclusive value="-1000000"/>
          <xs:maxInclusive value="1000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="dmDec" >
      <xs:simpleType>
        <xs:restriction base="xs:float">
          <xs:minInclusive value="-1000000"/>
          <xs:maxInclusive value="1000000"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <!-- Warning, above may need hand editing -->
    <!-- observe that min-max restrictions have -->
    <!-- not been added to this template yet -->
    <xs:element name="yd" >
      <xs:simpleType>
        <xs:restriction base="xs:int">
```

```

        <xs:minInclusive value="-17600000"/>
        <xs:maxInclusive value="17600000"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ydDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-17600000"/>
            <xs:maxInclusive value="17600000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="ft" >
    <xs:simpleType>
        <xs:restriction base="xs:int">
            <xs:minInclusive value="-52800000"/>
            <xs:maxInclusive value="52800000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="ftDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-52800000"/>
            <xs:maxInclusive value="52800000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="in" >
    <xs:simpleType>
        <xs:restriction base="xs:int">
            <xs:minInclusive value="-1000000"/>
            <xs:maxInclusive value="1000000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="inDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-1000000"/>
            <xs:maxInclusive value="1000000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="mi" >
    <xs:simpleType>
        <xs:restriction base="xs:short">
            <xs:minInclusive value="-10000"/>
            <xs:maxInclusive value="10000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="miDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-10000"/>
            <xs:maxInclusive value="10000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->

```

```

<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="km" >
  <xs:simpleType>
    <xs:restriction base="xs:short">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="kmDec" >
  <xs:simpleType>
    <xs:restriction base="xs:float">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
</xs:choice>
</xs:complexType>

```

#### 5.45 #45 EXT\_LRMS\_Height (Data Frame)

**Definition:** A vertical height expressed as either an altitude with reference to a vertical datum, or a vertical level from -127 to +127, where level 0 is the ground surface, or ground level of a structure.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```

Height ::= CHOICE {
  altdatum SEQUENCE {
    altitude      Distance,
    verticalDatum VerticalDatum OPTIONAL
  },
  verticalLevel   VerticalLevel
}

```

**XML Representation:**

```

<xs:complexType name="Height" >
  <xs:choice>
    <xs:element name="altdatum" >
      <xs:complexType>
        <xs:sequence>
          <xs:element name="altitude" type="Distance" />
          <xs:element name="verticalDatum" type="VerticalDatum" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="verticalLevel" type="VerticalLevel" />
  </xs:choice>
</xs:complexType>

```

#### 5.46 #46 EXT\_LRMS\_HorizontalDatum (Data Element)

**Definition:** The underlying horizontal geodetic datum for a geographic coordinate

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```

HorizontalDatum ::= ENUMERATED {
  wgs-84      (0),
  wgs-84egm-96 (1),
  nad83       (2),
  nad27       (3),
  ...
}

```

## XML Representation:

```
<xs:simpleType name="HorizontalDatum" >
  <xs:annotation>
    <xs:appinfo>
      wgs 84 (0)
      wgs 84egm 96 (1)
      nad83 (2)
      nad27 (3)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="3"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="wgs 84"/>
        <xs:enumeration value="wgs 84egm 96"/>
        <xs:enumeration value="nad83"/>
        <xs:enumeration value="nad27"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>
```

### 5.47 #47 EXT\_LRMS\_Latitude (Data Element)

**Definition:** The geographic latitude of a node, expressed in integer microdegrees, with reference to the horizontal datum specified by horizontalDatum.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Latitude ::= INTEGER (-90000000..90000000) -- microdegrees

#### XML Representation:

```
<xs:simpleType name="Latitude" >
  <xs:restriction base="xs:int">
    <xs:minInclusive value="-90000000"/>
    <xs:maxInclusive value="90000000"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.48 #48 EXT\_LRMS\_Longitude (Data Element)

**Definition:** The geographic longitude of a node, expressed in integer microdegrees, with reference to the horizontal datum specified by horizontalDatum.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Longitude ::= INTEGER (-180000000..180000000) -- microdegrees

#### XML Representation:

```
<xs:simpleType name="Longitude" >
  <xs:restriction base="xs:int">
    <xs:minInclusive value="-180000000"/>
    <xs:maxInclusive value="180000000"/>
  </xs:restriction>
</xs:simpleType>
```

**5.49 #49 EXT\_LRMS\_VerternalDatum (Data Element)**

**Definition:** The underlying vertical geodetic datum for a geographic coordinate.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
VerticalDatum ::= ENUMERATED {
    wgs-84 (0),
    navd (1),
    ...
}
```

**XML Representation:**

```
<xs:simpleType name="VerticalDatum" >
  <xs:annotation>
    <xs:appinfo>
      wgs 84 (0)
      navd (1)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="1"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="wgs 84"/>
        <xs:enumeration value="navd"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>
```

**5.50 #50 EXT\_LRMS\_VertecalLevel (Data Element)**

**Definition:** Vertical level specified as ordinal class expressed positive (above) or negative (below) the ground surface or ground level of a structure (VerticalLevel = 0).

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
VerticalLevel ::= INTEGER (-127..127)
```

**XML Representation:**

```
<xs:simpleType name="VerticalLevel" >
  <xs:restriction base="xs:byte">
    <xs:minInclusive value="-127"/>
  </xs:restriction>
</xs:simpleType>
```

**5.51 #51 Filters (Data Frame)**

**Definition:** Filters used in the subscription message process.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
Filters ::= SEQUENCE {
    dms-status-information BOOLEAN,
    -- subscription to DMS status information
    -- included true/false (yes/no)
    ess-status-information BOOLEAN,
    -- subscription to ESS status information
    -- included true/false (yes/no)
}
```

```
...
}
```

### XML Representation:

```
<xs:complexType name="Filters" >
  <xs:sequence>
    <xs:element name="dms-status-information" >
      <xs:simpleType>
        <xs:restriction base="xs:boolean"/>
      </xs:simpleType>
    </xs:element>
    <xs:element name="ess-status-information" >
      <xs:simpleType>
        <xs:restriction base="xs:boolean"/>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

## 5.52 #52 FreeText (Data Element)

**Definition:** A simple free-text field used in the message when structured information will not serve.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
FreeText ::= UTF8String (SIZE(1..500))
```

### XML Representation:

```
<xs:simpleType name="FreeText" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="500"/>
  </xs:restriction>
</xs:simpleType>
```

## 5.53 #53 InformationalText (Data Element)

**Definition:** Any set of ASCII characters up to 255.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
InformationalText ::= IA5String (SIZE(1..255))
```

### XML Representation:

```
<xs:simpleType name="InformationalText" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="255"/>
  </xs:restriction>
</xs:simpleType>
```

## 5.54 #54 LINK\_Direction\_code (Data Element)

**Definition:** The direction(s) of travel referenced on a link.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
Link-direction ::= ENUMERATED {
  any-other      (0),
  n               (1),
  ne             (2),
  e              (3),
  se             (4),
  s              (5),
  sw             (6),
```

```

w (7),
nw (8),
not-directional (9),
positive-direction (10),
negative-direction (11),
both-directions (12),
...
}

```

### XML Representation:

```

<xs:simpleType name="Link-direction" >
  <xs:annotation>
    <xs:appinfo>
      any other (0)
      n (1)
      ne (2)
      e (3)
      se (4)
      s (5)
      sw (6)
      w (7)
      nw (8)
      not directional (9)
      positive direction (10)
      negative direction (11)
      both directions (12)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="12"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="any other"/>
        <xs:enumeration value="n"/>
        <xs:enumeration value="ne"/>
        <xs:enumeration value="e"/>
        <xs:enumeration value="se"/>
        <xs:enumeration value="s"/>
        <xs:enumeration value="sw"/>
        <xs:enumeration value="w"/>
        <xs:enumeration value="nw"/>
        <xs:enumeration value="not directional"/>
        <xs:enumeration value="positive direction"/>
        <xs:enumeration value="negative direction"/>
        <xs:enumeration value="both directions"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

## 5.55 #55 LINK\_RouteDesignator\_identifier (Data Element)

**Definition:** County, State, or Federal route numbers with any associated alphabetic designators.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Link-route-designator ::= IA5String (SIZE(1..64)) -- Any set of alphanumeric characters up to 64

### XML Representation:

```

<xs:simpleType name="Link-route-designator" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>

```



```
</xs:simpleType>
```

### 5.56 #56 ORGANIZATION\_CenterIdentifier\_identifier (Data Element)

**Definition:** A unique identification number which identifies an organization's Center (transportation, emergency management, public safety, etc.) within a region.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Organization-center-identifier ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

#### XML Representation:

```
<xs:simpleType name="Organization-center-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.57 #57 ORGANIZATION\_Identifier\_identifier (Data Element)

**Definition:** A unique identifier for an organization within a region.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Organization-identifier ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

#### XML Representation:

```
<xs:simpleType name="Organization-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.58 #58 ORGANIZATION\_Name\_text (Data Element)

**Definition:** The organization's name.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Organization-name ::= IA5String (SIZE(1..128)) -- Any set of ASCII characters up to 128

#### XML Representation:

```
<xs:simpleType name="Organization-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.59 #59 ORGANIZATION\_SubOrganizationName\_text (Data Element)

**Definition:** The organization's "sub organization" - Department or Bureau, for example - name.

#### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

Organization-sub-organization-name ::= IA5String (SIZE(1..128)) -- Any set of ASCII characters up to 128

#### XML Representation:

```
<xs:simpleType name="Organization-sub-organization-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
```

## 5.60 #60 OrganizationInformationLong (Data Frame)

**Definition:** The long form of the organizational data frame. This is derived from several TMDD data frames and data elements and has been customized to reflect local DOT needs. The typical use of this frame is when requesting data about a device, or when building a reply with data about a device owned by a center. See also the **OrganizationInformationShort** entry which has a terse version also derived from the same TMDD data frames and elements. Typically these are used in pairs.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
OrganizationInformationLong ::= SEQUENCE {
  organization-id      Organization-identifier,
  organization-name     Organization-name,
  organization-sub-name Organization-sub-organization-name OPTIONAL,
  -- added by DCK, may not be wanted
  contactDetails       ContactDetails,
  ...
}
```

### XML Representation:

```
<xs:complexType name="OrganizationInformationLong" >
  <xs:sequence>
    <xs:element name="organization-id" type="Organization-identifier" />
    <xs:element name="organization-name" type="Organization-name" />
    <xs:element name="organization-sub-name" type="Organization-sub-organization-name"
minOccurs="0"/>
    <xs:element name="contactDetails" type="ContactDetails" />
  </xs:sequence>
</xs:complexType>
```

## 5.61 #61 OrganizationInformationShort (Data Frame)

**Definition:** The short form of the organizational data frame. This is derived from several TMDD data frames and data elements and has been customized to reflect local DOT needs. The typical use of this frame is when requesting data about a device, or when building a reply with data about a device owned by a center. See also the **OrganizationInformationLong** entry which has a verbose version also derived from the same TMDD data elements. Typically these are used in pairs.

### ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]

```
OrganizationInformationShort ::= SEQUENCE {
  organization-id      Organization-identifier,
  organization-name     Organization-name,
  ...
}
```

### XML Representation:

```
<xs:complexType name="OrganizationInformationShort" >
  <xs:sequence>
    <xs:element name="organization-id" type="Organization-identifier" />
    <xs:element name="organization-name" type="Organization-name" />
  </xs:sequence>
</xs:complexType>
```

### 5.62 #62 *ReturnAddress* (Data Element)

**Definition:** A URL indicating the subscriber callback message handler. Any set of ASCII characters up to 128.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

ReturnAddress ::= IA5String (SIZE(1..128))

**XML Representation:**

```
<xs:simpleType name="ReturnAddress" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.63 #63 *SECURITY\_Password\_text* (Data Element)

**Definition:** The unique password of a user to login.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Security-password ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

**XML Representation:**

```
<xs:simpleType name="Security-password" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.64 #64 *SECURITY\_UserName\_text* (Data Element)

**Definition:** The unique User name used to login to the system.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

Security-user-name ::= IA5String (SIZE(1..32)) -- Any set of alphanumeric characters up to 32

**XML Representation:**

```
<xs:simpleType name="Security-user-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
```

### 5.65 #65 *SubscriptionAction* (Data Frame)

**Definition:** A set of SubscriptionActionItem entries.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

SubscriptionAction ::= SEQUENCE (SIZE(1..10)) OF SubscriptionActionItem

**XML Representation:**

```
<xs:complexType name="SubscriptionAction" >
  <xs:sequence minOccurs="1" maxOccurs="10">
    <xs:element name="subscriptionAction-item" type="SubscriptionActionItem" />
  </xs:sequence>
```

</xs:complexType>

## 5.66 #66 SubscriptionActionItem (Data Element)

**Definition:** An ENUMERATED list of the subscription actions that a request message may take.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
SubscriptionActionItem ::= ENUMERATED {
    reserved (0),
    newSubscription (1),
    replaceSubscription (2),
    cancelSubscription (3),
    cancelAllPriorSubscriptions (4),
    ...
}
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionActionItem" >
  <xs:annotation>
    <xs:appinfo>
      reserved (0)
      newSubscription (1)
      replaceSubscription (2)
      cancelSubscription (3)
      cancelAllPriorSubscriptions (4)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="4"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="newSubscription"/>
        <xs:enumeration value="replaceSubscription"/>
        <xs:enumeration value="cancelSubscription"/>
        <xs:enumeration value="cancelAllPriorSubscriptions"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>
```

## 5.67 #67 SubscriptionCount (Data Element)

**Definition:** The nth time the publisher has sent content as part of a description to the subscriber.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
SubscriptionCount ::= INTEGER (1..4294967295)
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionCount" >
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>
```

## 5.68 #68 SubscriptionFrequency (Data Element)

**Definition:** The interval (period) in seconds

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
SubscriptionFrequency ::= INTEGER (1..4294967295)
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionFrequency" >
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>
```

**5.69 #69 SubscriptionID (Data Element)**

**Definition:** A unique name created by the subscriber for a subscription. Any set of ASCII characters up to 128

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
SubscriptionID ::= IA5String (SIZE(1..128))
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionID" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
```

**5.70 #70 SubscriptionName (Data Element)**

**Definition:** A textual name for the subscription for human readers *NOTE:* This is not in the XML fragment you sent to me.

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
SubscriptionName ::= IA5String (SIZE(1..128))
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionName" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
```

**5.71 #71 SubscriptionType (Data Frame)**

**Definition:** A collection of SubscriptionTypeItems.

**ASN.1 Representation: [Name, Data Type, Valid Value Rule or Body]**

```
SubscriptionType ::= SEQUENCE (SIZE(1..10)) OF SubscriptionTypeItem
```

**XML Representation:**

```
<xs:complexType name="SubscriptionType" >
  <xs:sequence minOccurs="1" maxOccurs="10">
    <xs:element name="subscriptionType-item" type="SubscriptionTypeItem" />
  </xs:sequence>
</xs:complexType>
```

**5.72 #72 SubscriptionTypeItem (Data Element)**

**Definition:** An ENUMERATED list of the subscription types that a request message may use.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
SubscriptionTypeItem ::= ENUMERATED {
    reserved      (0),
    oneTime       (1),
    periodic       (2),
    onChange      (3),
    ...
}
```

**XML Representation:**

```
<xs:simpleType name="SubscriptionTypeItem" >
  <xs:annotation>
    <xs:appinfo>
      reserved (0)
      oneTime (1)
      periodic (2)
      onChange (3)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="3"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="oneTime"/>
        <xs:enumeration value="periodic"/>
        <xs:enumeration value="onChange"/>
      </xs:restriction>
    </xs:simpleType >
  </xs:union>
</xs:simpleType>
```

**5.73 #73 TimeInterval (Data Element)**

**Definition:** A value for how often (in various units from years to seconds) an event should occur, such as updating a data file. Expressed in the native XML format when used.

**ASN.1 Representation:** [Name, Data Type, Valid Value Rule or Body]

```
TimeInterval ::= IA5String (SIZE(1..255)) -- Native XML format used
```

**XML Representation:**

```
<xs:simpleType name="TimeInterval">
  <xs:restriction base="xs:duration"/>
</xs:simpleType>
```

## Appendix A – Requirements Traceability Matrix

Legend:

DL - Dialog  
 MSG - Message  
 DE - Data Element  
 FN - Function

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
3.1.a	3.1 - Provide DMS Control to Remote Agencies		DL	DL_DMSSControlRequest
3.1.1	3.1.1 - Send DMS Control Request	The remote center shall be capable of sending a DMS control request message to the local center that controls a sign that a message is to be posted onto. The request shall include the following:	MSG	MSG_DMSSControlRequest
		· The ID of the receiving center		
		· The ID of the sending center		
		· The device ID of the DMS		
		· The unique request identifier assigned by the requesting center		
		· The security attribute (user name and password)		
		· The operator and agency name making the request		
		· The message number for the pre-defined message that is to be displayed, or		
		· The specific message to be displayed		
		· The message page flash time		
		· The priority of the message being requested		
		· The start time for the message		
		· The start date for the message		
		· The expiration time for the message		
		· The expiration date for the message		
		· Additional information/comments		
3.1.2	3.1.2 - Receive DMS Control Request	The local center shall be capable of accepting and processing valid DMS control requests to display a pre-defined or new text message from one or more authorized remote centers.	MSG	MSG_DMSSControlRequest

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
3.1.3	3.1.3 - Send DMS Control Response	The local center shall be capable of sending a response to the requesting center. The response to a DMS control request shall include the following:	MSG	MSG_DeviceControlSubscriptionResponse
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name in the request		
		· The name of the operator at the local center acting on the request		
		· The status of the request (implemented, queued, rejected)		
		· Additional information/comments		
3.1.4	3.1.4 - Receive DMS Control Response	The remote center shall be capable of receiving a response to a DMS control request.	MSG	MSG_DeviceControlSubscriptionResponse
3.1.b	3.1 - Provide DMS Control to Remote Agencies		DL	DL_DeviceCancelControlRequest
3.1.5	3.1.5 - Send DMS Cancel Control Request	The remote center shall be capable of sending a DMS cancel control request message if the center wishes to cancel a previous request to display a message. The request shall include the following:	MSG	MSG_DeviceCancelControlRequest
		· The ID of the receiving center		
		· The ID of the sending center		
		· The device ID of the DMS		
		· The unique request identifier assigned by the requesting center		
		· The security attribute (user name and password)		
		· The operator and agency name making the request		
		· Additional information/comments		
3.1.6	3.1.6 - Receive DMS Cancel Control Request	The local center shall be capable of accepting and processing valid DMS cancel control requests to terminate a message previously requested. The local center shall display any unexpired messages based on priority.	MSG	MSG_DeviceCancelControlRequest
3.1.7	3.1.7 - Send DMS Cancel Control Response	The local center shall be capable of sending a response to the requesting center to a DMS cancel control request. The response to a DMS cancel control request shall include the following:	MSG	MSG_DeviceControlSubscriptionResponse



## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name in the request		
		· The name of the operator at the local center acting on the cancellation request		
		· The status of the request (implemented, queued, rejected)		
		· Additional information/comments		
3.1.8	3.1.8 - Receive DMS Cancel Control Response	The remote center shall be capable of receiving a response to a DMS cancel control request.	MSG	MSG_DeviceControlSubscriptionResponse
3.2.a	3.2 - Provide DMS Inventory Information		DL	DL_DMSInventoryUpdateSubscriptionPublication
3.2.1	3.2.1 - Send DMS Inventory Update Subscription	The remote center shall be capable of sending a DMS inventory update subscription message to each center from which it wants to receive inventory update messages. The subscription shall include the following:	MSG	MSG_DeviceInventorySubscription
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier assigned by the requesting center		
		· The security attribute (user name and password)		
		· The operator and agency name making the subscription		
		· The start time of the subscription service		
		· The start date of the subscription service or 'ongoing' if subscribing to daily service		
		· The end time of the subscription service		
		· The end date of the subscription service or 'ongoing' if subscribing to daily service		
		· Subscription to DMS enabled/disabled		
		· Subscription to pre-defined messages enabled/disabled		
		· Subscription to environmental sensors enabled/disabled		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
3.2.2	3.2.2 - Receive DMS Inventory Update Subscription	The local center shall be capable of accepting and processing valid DMS inventory update subscriptions.	MSG	MSG_DeviceInventorySubscription
3.2.3	3.2.3 - Send DMS Inventory Update Subscription Response	The local center shall be capable of sending a response to the requesting center. The response to a DMS inventory update subscription shall include the following:	MSG	MSG_DeviceControlSubscriptionResponse
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name in the request		
		· The name of the operator at the local center acting on the request		
		· The subscription service ID		
3.2.4	3.2.4 - Receive DMS Inventory Update Subscription Response	The remote center shall be capable of receiving a response to a DMS inventory update subscription.	MSG	MSG_DeviceControlSubscriptionResponse
3.2.5	3.2.5 - Send DMS Inventory Update	The local center shall be capable of sending a DMS inventory update message to all subscribing remote centers upon DMS, pre-defined messages, and environmental sensors being added, removed, or changed. Messages shall only be sent between the start and end times of the subscription service. The update message shall include the following:	MSG	MSG_DMSInventoryPublication
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name making the update		
		· The contact information (name, phone number, pager, email address) for the owning center		
		· name		
		· phone number		
		· pager		
		· email address		
		(repeated for each pre-defined message)		
		· Type of update: addition, removal, change		
		· The message number for each pre-defined message that has been added, removed, or changed		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
		<ul style="list-style-type: none"> <li>The message content for the corresponding message number</li> </ul>		
		(repeated for each DMS)		
		<ul style="list-style-type: none"> <li>Type of update: addition, removal, change</li> </ul>		
		<ul style="list-style-type: none"> <li>The device ID of each DMS that has been added, removed, or changed</li> </ul>		
		<ul style="list-style-type: none"> <li>The location of the device (longitude and latitude)</li> </ul>		
		<ul style="list-style-type: none"> <li>The road name the sign is on</li> </ul>		
		<ul style="list-style-type: none"> <li>The direction of travel the sign faces</li> </ul>		
		<ul style="list-style-type: none"> <li>The sign make and model</li> </ul>		
		<ul style="list-style-type: none"> <li>The sign type (VMS, CMS, BOS) and technology (LED, flip-disk, fiber optic)</li> </ul>		
		<ul style="list-style-type: none"> <li>The number of lines supported, if any</li> </ul>		
		<ul style="list-style-type: none"> <li>The number of characters supported, if any</li> </ul>		
		<ul style="list-style-type: none"> <li>The number of scroll pages supported, if any</li> </ul>		
		(repeated for each sensor)		
		<ul style="list-style-type: none"> <li>Type of update: addition, removal, change</li> </ul>		
		<ul style="list-style-type: none"> <li>The device ID of each environmental sensor that has been added, removed, or changed</li> </ul>		
		<ul style="list-style-type: none"> <li>The location of the device (longitude and latitude)</li> </ul>		
		<ul style="list-style-type: none"> <li>The elevation of the sensor</li> </ul>		
		<ul style="list-style-type: none"> <li>The road name the sensor is on</li> </ul>		
		<ul style="list-style-type: none"> <li>The sensor description/purpose</li> </ul>		
3.2.6	3.2.6 - Receive DMS Inventory Update	The remote center shall be capable of receiving a DMS inventory update message.	MSG	MSG_DMSInventoryPublication
3.2.c	3.2 - Provide DMS Inventory Information		DL	DL_DMSInventoryRequest
3.2.7	3.2.7 - Send DMS Inventory Request	The remote center shall be capable of sending a DMS inventory request message to other centers upon connecting to the C2C network. The request shall include the following:	MSG	MSG_DeviceInventoryRequest
		<ul style="list-style-type: none"> <li>The ID of the receiving center</li> </ul>		
		<ul style="list-style-type: none"> <li>The ID of the sending center</li> </ul>		
		<ul style="list-style-type: none"> <li>The unique request identifier assigned by the requesting center</li> </ul>		
		<ul style="list-style-type: none"> <li>The security attribute (user name and password)</li> </ul>		
		<ul style="list-style-type: none"> <li>The operator and agency name making the request</li> </ul>		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
3.2.8	3.2.8 - Receive DMS Inventory Request	The local center shall be capable of accepting and processing valid DMS inventory requests.	MSG	MSG_DeviceInventoryRequest
3.2.9	3.2.9 - Send DMS Inventory Response	The local center shall be capable of sending a response to the requesting center. The response to a DMS inventory request shall include the following:	MSG	MSG_DMSInventoryResponse
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name in the request		
		· The name of the operator at the local center acting on the request		
		· The contact information (name, phone number, pager, email address) for the owning center		
		· name		
		· phone number		
		· pager		
		· email address		
		(repeated for each pre-defined message)		
		· The message number for each pre-defined message		
		· The message content for the corresponding message number		
		(repeated for each DMS)		
		· The device ID of each DMS		
		· The location of the device (longitude and latitude)		
		· The road name the sign is on		
		· The direction of travel the sign faces		
		· The sign make and model		
		· The sign type (VMS, CMS, BOS) and technology (LED, flip-disk, fiber optic)		
		· The number of lines supported, if any		
		· The number of characters supported, if any		
		· The number of scroll pages supported, if any		
		(repeated for each sensor)		
		· The device ID of each environmental sensor		
		· The location of the device (longitude and latitude)		
		· The elevation of the sensor		
		· The road name the sensor is on		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
		<ul style="list-style-type: none"> <li>The sensor description/purpose</li> </ul>		
3.2.10	3.2.10 - Receive DMS Inventory Response	The remote center shall be capable of receiving a response to a DMS inventory request.	MSG	MSG_DMSInventoryResponse
3.3.a	3.3 - Provide DMS Status Information		DL	DL_DMSStatusRequest
3.3.1	3.3.1 - Send DMS Status Information Request	The remote center shall be capable of sending a DMS status information request message to the local center that controls a sign that is being queried. The request shall include the following:	MSG	MSG_DeviceStatusRequest
		<ul style="list-style-type: none"> <li>The ID of the receiving center</li> </ul>		
		<ul style="list-style-type: none"> <li>The ID of the sending center</li> </ul>		
		<ul style="list-style-type: none"> <li>The device ID of the DMS</li> </ul>		
		<ul style="list-style-type: none"> <li>The unique request identifier assigned by the requesting center</li> </ul>		
		<ul style="list-style-type: none"> <li>The security attribute (user name and password)</li> </ul>		
		<ul style="list-style-type: none"> <li>user name</li> </ul>		
		<ul style="list-style-type: none"> <li>password</li> </ul>		
		<ul style="list-style-type: none"> <li>The operator and agency name making the request</li> </ul>		
3.3.2	3.3.2 - Receive DMS Status Information Request	The local center shall be capable of sending a response to the requesting center. The response to a DMS status information request shall include the following:	MSG	MSG_DeviceStatusRequest
		<ul style="list-style-type: none"> <li>The ID of the receiving center</li> </ul>		
		<ul style="list-style-type: none"> <li>The ID of the sending center</li> </ul>		
		<ul style="list-style-type: none"> <li>The unique request identifier</li> </ul>		
		<ul style="list-style-type: none"> <li>The operator and agency name in the request</li> </ul>		
		<ul style="list-style-type: none"> <li>The name of the operator at the local center acting on the request</li> </ul>		
		<ul style="list-style-type: none"> <li>The operational status of the device (on, off, failed)</li> </ul>		
		<ul style="list-style-type: none"> <li>The message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The name of the agency which put the message on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The priority of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The start time of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The start date of the message currently being displayed on the sign</li> </ul>		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
		<ul style="list-style-type: none"> <li>The expiration time of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The expiration date of the message currently being displayed on the sign</li> </ul>		
3.3.3	3.3.3 - Send DMS Status Information Response	The local center shall be capable of sending a response to the requesting center. The response to a DMS status information request shall include the following:	MSG	MSG_DMSStatusResponse
		<ul style="list-style-type: none"> <li>The ID of the receiving center</li> </ul>		
		<ul style="list-style-type: none"> <li>The ID of the sending center</li> </ul>		
		<ul style="list-style-type: none"> <li>The unique request identifier</li> </ul>		
		<ul style="list-style-type: none"> <li>The operator and agency name in the request</li> </ul>		
		<ul style="list-style-type: none"> <li>The name of the operator at the local center acting on the request</li> </ul>		
		<ul style="list-style-type: none"> <li>The operational status of the device (on, off, failed)</li> </ul>		
		<ul style="list-style-type: none"> <li>The message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The name of the agency which put the message on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The priority of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The start time of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The start date of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The expiration time of the message currently being displayed on the sign</li> </ul>		
		<ul style="list-style-type: none"> <li>The expiration date of the message currently being displayed on the sign</li> </ul>		
3.3.4	3.3.4 - Receive DMS Status Information Response	The remote center shall be capable of receiving a response to a DMS status information request.	MSG	MSG_DMSStatusResponse
3.3.b	3.3 - Provide DMS Status Information		DL	DL_DMSStatusSubscriptionPublication
3.3.5	3.3.5 - Send DMS Status Information Subscription	The remote center shall be capable of sending a DMS status information subscription message to each center from which it wants to receive status update messages. The subscription shall include the following:	MSG	MSG_DeviceStatusSubscription
		<ul style="list-style-type: none"> <li>The ID of the receiving center</li> </ul>		

## New York State ITS Standards Specification Development Guide

Requirement Id	Requirements Document Section	Requirement	Data Concept Type	Data Concept Name
		· The ID of the sending center		
		· The unique request identifier assigned by the requesting center		
		· The security attribute (user name and password)		
		· The operator and agency name making the subscription		
		· The start time of the subscription service		
		· The start date of the subscription service or 'ongoing' if subscribing to daily service		
		· The end time of the subscription service		
		· The end date of the subscription service or 'ongoing' if subscribing to daily service		
3.3.6	3.3.6 - Receive DMS Status Information Subscription	The local center shall be capable of accepting and processing valid DMS status information subscriptions.	MSG	MSG_DeviceStatusSubscription
3.3.7	3.3.7 - Send DMS Status Information Subscription Response	The local center shall be capable of sending a response to the requesting center. The response to a DMS status information subscription shall include the following:	MSG	MSG_DMSStatusResponse
		· The ID of the receiving center		
		· The ID of the sending center		
		· The unique request identifier		
		· The operator and agency name in the request		
		· The name of the operator at the local center acting on the request		
		· The subscription service ID		
3.3.8	3.3.8 - Receive DMS Status Information Subscription Response	The remote center shall be capable of receiving a response to a DMS status information subscription.	MSG	MSG_DMSStatusResponse





## Appendix B - Web Services Description Language Listing

This section provides the formal Web Services Description Language document for the subject project. The WSDL is based on the text and examples provided in the NTCIP 2306 standard. The WSDL is also available in XML electronic format at <http://www.someplace.com/c2cxml/specification>.

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="CenterServices" targetNamespace="http://center-services"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:tns="http://center-services"
  xmlns:nystate="http://www.NYState-Sample"
  xmlns:ftp="http://schemas.ntcip.org/wsdl/ftp/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
>
<!-- TYPES -->
<types>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:import namespace="http://www.NYState-Sample" schemaLocation="nystate.xsd"/>
  </xs:schema>
</types>
<!-- MESSAGES -->

<message name="MSG_CancelSubscriptionRequest">
  <part name="message" element="nystate:cancelSubscriptionRequest"/>
</message>
<message name="MSG_DeviceCancelControlRequest">
  <part name="message" element="nystate:deviceCancelControlRequest"/>
</message>
<message name="MSG_DeviceControlSubscriptionResponse">
  <part name="message" element="nystate:deviceControlSubscriptionResponse"/>
</message>
<message name="MSG_DeviceInventoryRequest">
  <part name="message" element="nystate:deviceInventoryRequest"/>
</message>
<message name="MSG_DeviceInventorySubscription">
  <part name="message" element="nystate:deviceInventorySubscription"/>
</message>
<message name="MSG_DeviceStatusRequest">
  <part name="message" element="nystate:deviceStatusRequest"/>
</message>
<message name="MSG_DeviceStatusSubscription">
  <part name="message" element="nystate:deviceStatusSubscription"/>
</message>
<message name="MSG_DMSTControlRequest">
  <part name="message" element="nystate:dmsControlRequest"/>
</message>
<message name="MSG_DMSTInventoryResponse">
  <part name="message" element="nystate:dmsInventoryResponse"/>
</message>
<message name="MSG_DMSTInventoryPublication">
  <part name="message" element="nystate:dmsInventoryPublication"/>
</message>
<message name="MSG_DMSTStatusResponse">
  <part name="message" element="nystate:dmsStatusResponse"/>
</message>
<message name="MSG_DMSTStatusPublication">
  <part name="message" element="nystate:dmsStatusPublication"/>
</message>

<!-- ***** -->
<!-- PORT TYPE - nystateSoapHttpService / SOAP / HTTP -->
<portType name="nystateSoapHttpServicePortType">
```

```

<operation name = "OP_CancelSubscriptionRequest">
  <input message="tns:MSG_CancelSubscriptionRequest"/>
  <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
</operation>
<operation name = "OP_DMSSControlRequest">
  <input message="tns:MSG_DMSSControlRequest"/>
  <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
</operation>
<operation name = "OP_DMSSInventoryRequest">
  <input message="tns:MSG_DeviceInventoryRequest"/>
  <output message="tns:MSG_DMSSInventoryResponse"/>
</operation>
<operation name = "OP_DMSSStatusRequest">
  <input message="tns:MSG_DeviceStatusRequest"/>
  <output message="tns:MSG_DMSSStatusResponse"/>
</operation>
<operation name = "OP_DeviceCancelControlRequest">
  <input message="tns:MSG_DeviceCancelControlRequest"/>
  <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
</operation>
<operation name = "OP_DeviceInventoryUpdateSubscription">
  <input message="tns:MSG_DeviceInventorySubscription"/>
  <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
</operation>
<operation name = "OP_DeviceStatusUpdateSubscription">
  <input message="tns:MSG_DeviceStatusSubscription"/>
  <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
</operation>
</portType>

<!-- ***** -->
<!-- PORT TYPE - nystateSubscriberCallbackService / SOAP / HTTP -->
<portType name="nystateSubscriberCallbackServicePortType">
  <operation name = "OP_SubscriberDMSInventoryUpdate">
    <input message="tns:MSG_DMSSInventoryPublication"/>
    <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
  </operation>
  <operation name = "OP_SubscriberDMSStatusUpdate">
    <input message="tns:MSG_DMSSStatusPublication"/>
    <output message="tns:MSG_DeviceControlSubscriptionResponse"/>
  </operation>
</portType>

<!-- ***** -->
<!-- BINDING - nystateSoapHttpService / SOAP / HTTP -->
<binding name="nystateSoapHttpServiceBinding"
  type="tns:nystateSoapHttpServicePortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name = "OP_CancelSubscriptionRequest">
    <soap:operation soapAction="dummy.cgi" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name = "OP_DMSSControlRequest">
    <soap:operation soapAction="dummy.cgi" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name = "OP_DMSSInventoryRequest">
    <soap:operation soapAction="dummy.cgi" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
  </operation>

```

```

        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
    <operation name = "OP_DMSStatusRequest">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
    <operation name = "OP_DeviceCancelControlRequest">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
    <operation name = "OP_DeviceInventoryUpdateSubscription">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
    <operation name = "OP_DeviceStatusUpdateSubscription">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
</binding>

<!-- ***** -->
<!-- BINDING - nystateSubscriberCallbackService / SOAP / HTTP -->
<binding name="nystateSubscriberCallbackServiceBinding"
    type="tns:nystateSubscriberCallbackServicePortType">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name = "OP_SubscriberDMSInventoryUpdate">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
    <operation name = "OP_SubscriberDMSStatusUpdate">
        <soap:operation soapAction="dummy.cgi" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
</binding>

<!-- SERVICE - nystateSoapHttpService / SOAP / HTTP -->
<service name="nystateSoapHttpService">
    <port name="nystateSoapHttpServicePort"
        binding="tns:nystateSoapHttpServiceBinding">

```

```
        <soap:address location="http://myc2c.system.com/c2cxml/" />
    </port>
</service>

<!-- SERVICE - nystateSubscriberCallbackService / SOAP / HTTP -->
<service name="nystateSubscriberCallbackService">
    <port name="nystateSubscriberCallbackServicePort"
        binding="tns:nystateSubscriberCallbackServiceBinding">
        <soap:address location="http://myc2c.subscriber.com/c2cxml" />
    </port>
</service>

</definitions>
```

## Appendix C - XML Schema Listing

This section describes the XML schema that shall be used to validate the XML messages exchanged between NYState ITS Project C2C systems. The XML Schemas are also available in XML electronic format at <http://www.someplace.com/c2cxml/specification>.

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="http://www.NYState-Sample"
  xmlns="http://www.NYState-Sample"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="unqualified"
  attributeFormDefault="unqualified"
  version="DRAFT">

  <!-- #1 AuthorizationSet (Data Frame) -->
  <!-- Definition: A data frame which contains the authorization information provided by a user as
  credentials for a message. Typically this information is validated against an X.509 certificate
  using XMLSEC methods. The contents of this data frame should be suppressed if the message is to
  be relayed to other down stream machines after the authorization has completed. The entire
  message should be deleted if the authorization fails. -->
  <xs:complexType name="AuthorizationSet" >
    <xs:sequence>
      <xs:element name="user-id" type="Security-user-name" />
      <xs:element name="password" type="Security-password" />
      <xs:element name="hashtime" type="DateTimePair" />
      <xs:element name="hash" >
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="10"/>
            <xs:maxLength value="20"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- #2 BroadcastAlerts (Data Frame) -->
  <!-- Definition: A collection of BroadcastAlertsItem items -->

  <xs:complexType name="BroadcastAlerts" >
    <xs:sequence minOccurs="1" maxOccurs="2">
      <xs:element name="broadcastAlert" type="BroadcastAlertsItem" />
    </xs:sequence>
  </xs:complexType>

  <!-- #3 BroadcastAlertsItem (Data Element) -->
  <!-- Definition: An enumerated list of items. -->

  <xs:simpleType name="BroadcastAlertsItem" >
    <xs:annotation>
      <xs:appinfo>
        reserved (0)
        broadcastAlertsAccepted (1)
        broadcastAlertsNotAccepted (2)
      </xs:appinfo>
    </xs:annotation>
    <xs:union>
      <xs:simpleType>
        <xs:restriction base="xs:unsignedInt">
          <xs:minInclusive value="0"/>
          <xs:maxInclusive value="2"/>
        </xs:restriction>
      </xs:simpleType>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="reserved"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
```

```

        <xs:enumeration value="broadcastAlertsAccepted"/>
        <xs:enumeration value="broadcastAlertsNotAccepted"/>
    </xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>
<!-- #4 C2C Message Publication (Data Frame) -->
<!-- Definition: This is a data frame used in the response to the subscription process. -->

<xs:complexType name="C2cMessagePublication" >
    <xs:sequence>
        <xs:element name="informationalText" type="InformationalText" minOccurs="0"/>
        <xs:element name="subscriptionID" type="SubscriptionID" />
        <xs:element name="subscriptionName" type="SubscriptionName" minOccurs="0"/>
        <xs:element name="subscriptionFreq" type="SubscriptionFrequency" minOccurs="0"/>
        <xs:element name="subscriptionCount" type="SubscriptionCount" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<!-- #5 C2C Message Subscription (Data Frame) -->
<!-- Definition: This is a data frame used in the subscription process. -->

<xs:complexType name="C2cMessageSubscription" >
    <xs:sequence>
        <xs:element name="informationalText" type="InformationalText" minOccurs="0"/>
        <xs:element name="returnAddress" type="ReturnAddress" />
        <xs:element name="subscriptionAction" type="SubscriptionAction" />
        <xs:element name="subscriptionType" type="SubscriptionType" />
        <xs:element name="subscriptionID" type="SubscriptionID" />
        <xs:element name="subscriptionName" type="SubscriptionName" minOccurs="0"/>
        <xs:element name="subscriptionTimeFrame" type="SubscriptionTimeFrame" minOccurs="0"/>
        <xs:element name="subscriptionFrequency" type="SubscriptionFrequency" />
        <xs:element name="broadcastAlerts" type="BroadcastAlerts" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<!-- #6 C2C_MessageReceipt (Data Frame) -->
<!-- Definition: AAA An empty Definition field. -->

<xs:complexType name="C2cMessageReceipt" >
    <xs:sequence>
        <xs:element name="informationalText" type="InformationalText" />
    </xs:sequence>
</xs:complexType>

<!-- #7 C2C_SubscriptionTimeFrame (Data Frame) -->
<!-- Definition: Time frame during which (the) subscriber requests that (the) publication be
active. -->

<xs:complexType name="SubscriptionTimeFrame" >
    <xs:sequence>
        <xs:element name="start" type="DateTimePair" />
        <xs:element name="end" type="DateTimePair" />
    </xs:sequence>
</xs:complexType>

<!-- #8 CancelSubscriptionRequest (Message) -->
<!-- Definition: C2C - Cancel Subscription Request Message, Based on Device Cancel Control
Request Messages for DMS and Signals. -->

<xs:element name="cancelSubscriptionRequest" type="CancelSubscriptionRequest"/>
<xs:complexType name="CancelSubscriptionRequest" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>

<!-- #9 CONTACT_EmailAddress_text (Data Element) -->
<!-- Definition: The e-mail address of a person at an organizatin to contact regarding a roadway
event. -->

```

```

<xs:simpleType name="Contact-email-address" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>

<!-- #10 CONTACT_Identifier_identifier (Data Element) -->
<!-- Definition: A unique identifier for a contact person as assigned by the organization. -->

<xs:simpleType name="Contact-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #11 CONTACT_PersonName_text (Data Element) -->
<!-- Definition: The name or identifier of a person at an organization. -->

<xs:simpleType name="Contact-person-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>

<!-- #12 CONTACT_PersonTitle_text (Data Element) -->
<!-- Definition: The title of a person at an organization. -->

<xs:simpleType name="Contact-person-title" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>

<!-- #13 CONTACT_PhoneAlternate_text (Data Element) -->
<!-- Definition: An alternate phone number (e.g., home phone number) of a person at an organization. -->

<xs:simpleType name="Contact-phone-alternate" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #14 CONTACT_PhoneNumber_text (Data Element) -->
<!-- Definition: The telephone number of a person at an organization. -->

<xs:simpleType name="Contact-phone-number" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #15 ContactDetails (Data Frame) -->
<!-- Definition: A collection of contact data elements defined by TMDD and refined for use by DOT -->

<xs:complexType name="ContactDetails" >
  <xs:sequence>
    <xs:element name="contact-id" type="Contact-identifier" />
    <xs:element name="person-name" type="Contact-person-name" minOccurs="0"/>
    <xs:element name="person-title" type="Contact-person-title" minOccurs="0"/>
    <xs:element name="phone-number" type="Contact-phone-number" minOccurs="0"/>
    <xs:element name="phone-alternate" type="Contact-phone-alternate" minOccurs="0"/>
    <xs:element name="email-address" type="Contact-email-address" minOccurs="0"/>
  </xs:sequence>

```

```

</xs:complexType>
<!-- #16 DateTimePair (Data Frame) -->
<!-- Definition: A representation of date, time, and time zone in the standard XML formats. -->

<xs:simpleType name="DateTimePair">
  <xs:restriction base="xs:dateTime"/>
</xs:simpleType>

<!-- Native XML formats are used for date and time -->
<!-- #17 DEVICE_AcknowledgeControl_code (Data Element) -->
<!-- Definition: Acknowledgement of request command from one TMC to another for device action. --
>

<xs:simpleType name="Device-acknowledge-control" >
  <xs:annotation>
    <xs:appinfo>
      control acknowledged (0)
      device available (1)
      requested changes completed (2)
      control rejected (3)
      device in use (4)
      device off line (5)
      request cancelled (6)
      request rejected invalid command parameters (7)
      request rejected insufficient privileges of the requesting operator (8)
      request queued (9)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="9"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="control acknowledged"/>
        <xs:enumeration value="device available"/>
        <xs:enumeration value="requested changes completed"/>
        <xs:enumeration value="control rejected"/>
        <xs:enumeration value="device in use"/>
        <xs:enumeration value="device off line"/>
        <xs:enumeration value="request cancelled"/>
        <xs:enumeration value="request rejected invalid command parameters"/>
        <xs:enumeration value="request rejected insufficient privileges of the requesting
operator"/>
        <xs:enumeration value="request queued"/>
      </xs:restriction>
    </xs:simpleType >
  </xs:union>
</xs:simpleType>
<!-- #18 DEVICE_CommandEndTime_utc (Data Element) -->
<!-- Definition: Expiration time of the command requested for the device control -->

<xs:simpleType name="Device-command-end-time" >
  <xs:restriction base="xs:int"/>
</xs:simpleType>

<!-- #19 DEVICE_CommandRequestPriority_number (Data Element) -->
<!-- Definition: A code to indicate universal concept of priority for all command requests. If
the request is accepted and the action is posted, then the current state has that priority. -->

<xs:simpleType name="Device-command-request-priority" >
  <xs:restriction base="xs:unsignedByte">
    <xs:minInclusive value="1"/>
    <xs:maxInclusive value="10"/>
  </xs:restriction>
</xs:simpleType>

<!-- #20 DEVICE_Identifier_identifier (Data Element) -->

```



```

<!-- Definition: A unique alphanumeric device identifier. -->

<xs:simpleType name="Device-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
<!-- #21 DEVICE_OperationalStatus_code (Data Element) -->
<!-- Definition: Operational status of a device (e.g., on, off, etc.). -->

<xs:simpleType name="Device-operational-status" >
  <xs:annotation>
    <xs:appinfo>
      on (1)
      off (2)
      in service (3)
      out of service (4)
      unavailable (5)
      unknown (6)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="6"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="on"/>
        <xs:enumeration value="off"/>
        <xs:enumeration value="in service"/>
        <xs:enumeration value="out of service"/>
        <xs:enumeration value="unavailable"/>
        <xs:enumeration value="unknown"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

<!-- #22 DEVICE_RequestIdentifier_identifier (Data Element) -->
<!-- Definition: A unique sequence number generated by the requesting center that uniquely
identifies the control request within the requesting center. -->

<xs:simpleType name="Device-request-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #23 DEVICE_Type_code (Data Element) -->
<!-- Definition: A code which specifies the type of device. -->

<xs:simpleType name="Device-type" >
  <xs:annotation>
    <xs:appinfo>
      detector (1)
      cctv camera (2)
      dynamic message sign (3)
      environmental sensor station (4)
      gate (5)
      highway advisory radio (6)
      lane control signal (7)
      ramp meter (8)
      signal controller (9)
      signal section (10)
    </xs:appinfo>
  </xs:annotation>

```

```

<xs:union>
  <xs:simpleType>
    <xs:restriction base="xs:unsignedInt">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="10"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="detector"/>
      <xs:enumeration value="cctv camera"/>
      <xs:enumeration value="dynamic message sign"/>
      <xs:enumeration value="environmental sensor station"/>
      <xs:enumeration value="gate"/>
      <xs:enumeration value="highway advisory radio"/>
      <xs:enumeration value="lane control signal"/>
      <xs:enumeration value="ramp meter"/>
      <xs:enumeration value="signal controller"/>
      <xs:enumeration value="signal section"/>
    </xs:restriction>
  </xs:simpleType>
</xs:union>
</xs:simpleType>
<!-- #24 DeviceArchiveStatusSubscription (Message) -->
<!-- Definition: C2C - Device Archive Status Information Subscription -->

<xs:element name="deviceArchiveStatusSubscription" type="DeviceArchiveStatusSubscription"/>
<xs:complexType name="DeviceArchiveStatusSubscription" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="request-id" type="Device-request-identifier" />
    <xs:element name="command-start-time" type="DateTimePair" />
    <xs:element name="command-end-time" type="DateTimePair" />
    <xs:element name="archive-frequency" type="TimeInterval" />
    <xs:element name="request-send-time" type="Time" />
    <xs:element name="request-send-date" type="Date" />
    <xs:element name="filters" type="Filters" />
  </xs:sequence>
</xs:complexType>

<!-- #25 DeviceCancelControlRequest (Message) -->
<!-- Definition: C2C - Device Cancel Control Request Message, Based on TMDD Device Control
Response Messages for DMS and Signals -->

<xs:element name="deviceCancelControlRequest" type="DeviceCancelControlRequest"/>
<xs:complexType name="DeviceCancelControlRequest" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="device-type" type="Device-type" />
    <xs:element name="device-id" type="Device-identifier" />
    <xs:element name="request-id" type="Device-request-identifier" />
    <xs:element name="freeText" type="FreeText" />
  </xs:sequence>
</xs:complexType>

<!-- #26 DeviceControlSubscriptionResponse (Message) -->
<!-- Definition: C2C - Device Message Response
Based on TMDD Device Control Response Messages for DMS and Signals Also used to acknowledge the
following: a Device Inventory Update Subscriptions for DMS and Signals, Device Inventory Update
Subscriptions, Device Status Update Subscriptions, Subscription Cancellations, Archival
Subscription Cancellations. -->

<xs:element name="deviceControlSubscriptionResponse" type="DeviceControlSubscriptionResponse"/>
<xs:complexType name="DeviceControlSubscriptionResponse" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationLong" />
    <xs:element name="organization-requesting" type="OrganizationInformationShort" />

```

```

        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="request-id" type="Device-request-identifier" />
        <xs:element name="device-id" type="Device-identifier" />
        <xs:element name="request-status" type="Device-acknowledge-control" />
        <xs:element name="freeText" type="FreeText" />
    </xs:sequence>
</xs:complexType>

<!-- #27 DeviceInventoryRequest (Message) -->
<!-- Definition: C2C - Device Inventory Request and Subscription Message, Based on TMDD DMS and
Intersection (Signal) Inventory Request Messages -->

<xs:element name="deviceInventoryRequest" type="DeviceInventoryRequest"/>
<xs:complexType name="DeviceInventoryRequest" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="device-type" type="Device-type" />
        <xs:element name="device-list" >
            <xs:complexType>
                <xs:sequence minOccurs="1" maxOccurs="100000">
                    <xs:element name="device" type="Device-identifier" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>

<!-- #28 DeviceInventorySubscription (Message) -->
<!-- Definition: C2C - Device Inventory Subscription Message. Based on Device Inventory Request
Message Content of message enclosed in SOAP Body -->

<xs:element name="deviceInventorySubscription" type="DeviceInventorySubscription"/>
<xs:complexType name="DeviceInventorySubscription" >
    <xs:sequence>
        <xs:element name="c2cMessageSubscription" type="C2cMessageSubscription" />
        <xs:element name="deviceInventoryRequest" type="DeviceInventoryRequest" />
    </xs:sequence>
</xs:complexType>

<!-- #29 DeviceStatusRequest (Message) -->
<!-- Definition: C2C - Device Status Information Request Message
Based on TMDD DMS and Intersection (Signal) Status Request Messages -->
<xs:element name="deviceStatusRequest" type="DeviceStatusRequest"/>
<xs:complexType name="DeviceStatusRequest" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="device-type" type="Device-type" />
        <xs:element name="device-list" >
            <xs:complexType>
                <xs:sequence minOccurs="1" maxOccurs="100000">
                    <xs:element name="device" type="Device-identifier" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>

<!-- #30 DeviceStatusSubscription (Message) -->
<!-- Definition: C2C - Device Status Subscription Message, Based on Device Status Request Message
Content of message enclosed in SOAP Body -->

<xs:element name="deviceStatusSubscription" type="DeviceStatusSubscription"/>
<xs:complexType name="DeviceStatusSubscription" >
    <xs:sequence>
        <xs:element name="c2cMessageSubscription" type="C2cMessageSubscription" />
        <xs:element name="deviceStatusRequest" type="DeviceStatusRequest" />
    </xs:sequence>
</xs:complexType>

```

```

        </xs:sequence>
    </xs:complexType>

<!-- #31 DMSControlRequest (Message) -->
<!-- Definition: C2C - DMS Control Request Message, Based on TMDD DMS Control Request Message -->

<xs:element name="dMSControlRequest" type="DMSControlRequest"/>
<xs:complexType name="DMSControlRequest" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationShort" />
        <xs:element name="organization-requesting" type="OrganizationInformationLong" />
        <xs:element name="authorization" type="AuthorizationSet" />
        <xs:element name="device-id" type="Device-identifier" />
        <xs:element name="request-id" type="Device-request-identifier" />
        <xs:element name="dms-beacon-control" type="DmsMessageMultiString" />
        <xs:element name="dms-message" type="DmsMessageMultiString" />
        <xs:element name="message-number" type="DmsMessageNumber" />
        <xs:element name="command-request-priority" type="Device-command-request-priority" />
        <xs:element name="command-start-time" type="DateTimePair" />
        <xs:element name="command-end-time" type="DateTimePair" />
        <xs:element name="freeText" type="FreeText" />
    </xs:sequence>
</xs:complexType>

<!-- #32 DMSDeviceStatus (Data Frame) -->
<!-- Definition: In TMDD this is documented a message, here is it used as a data frame. -->

<xs:element name="dMSDeviceStatus" type="DMSDeviceStatus"/>
<xs:complexType name="DMSDeviceStatus" >
    <xs:sequence>
        <xs:element name="device-id" type="Device-identifier" />
        <xs:element name="dms-device-status" type="Device-operational-status" />
        <xs:element name="dms-current-message" type="DmsMessageMultiString" />
        <xs:element name="message-time-remaining" type="DmsMessageTimeRemaining"
minOccurs="0"/>
        <xs:element name="last-comm-time" type="DateTimePair" />
    </xs:sequence>
</xs:complexType>

<!-- #33 DMSInventoryPublication (Message) -->
<!-- Definition: C2C - DMS Inventory Update Publication Message
Based on C2C DMS Inventory Response Message -->

<xs:element name="dMSInventoryPublication" type="DMSInventoryPublication"/>
<xs:complexType name="DMSInventoryPublication" >
    <xs:sequence>
        <xs:element name="c2cMessagePublication" type="C2cMessagePublication" />
        <xs:element name="dMSInventory" type="DMSInventoryResponse" />
    </xs:sequence>
</xs:complexType>

<!-- #34 DMSInventoryResponse (Message) -->
<!-- Definition: C2C - DMS Inventory Response Message, Based on TMDD DMS Inventory Message and
the DMSDeviceStatus data frame. -->

<xs:element name="dMSInventoryResponse" type="DMSInventoryResponse"/>
<xs:complexType name="DMSInventoryResponse" >
    <xs:sequence>
        <xs:element name="organization-owning" type="OrganizationInformationLong" />
        <xs:element name="organization-requesting" type="OrganizationInformationShort" />
        <xs:element name="devices" >
            <xs:complexType>
                <xs:sequence minOccurs="0" maxOccurs="10000">
                    <xs:element name="device" type="DMSDeviceStatus" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="request-id" type="Device-request-identifier" />
    </xs:sequence>
</xs:complexType>

```

```

<!-- #35 DMSStatusPublication (Message) -->
<!-- Definition: C2C - DMS Status Information Response Message used in replying to a subscription
request. -->

<xs:element name="dMSStatusPublication" type="DMSStatusPublication"/>
<xs:complexType name="DMSStatusPublication" >
  <xs:sequence>
    <xs:element name="c2cMessagePublication" type="C2cMessagePublication" />
    <xs:element name="dMSStatus" type="DMSStatusResponse" />
  </xs:sequence>
</xs:complexType>

<!-- #36 DMSStatusResponse (Message) -->
<!-- Definition: C2C - DMS Status Information Response Message -->

<xs:element name="dMSStatusResponse" type="DMSStatusResponse"/>
<xs:complexType name="DMSStatusResponse" >
  <xs:sequence>
    <xs:element name="organization-owning" type="OrganizationInformationShort" />
    <xs:element name="organization-requesting" type="OrganizationInformationLong" />
    <xs:element name="authorization" type="AuthorizationSet" />
    <xs:element name="devices" >
      <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs="10000">
          <xs:element name="device" type="DMSDeviceStatus" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="request-id" type="Device-request-identifier" />
  </xs:sequence>
</xs:complexType>

<!-- #37 EXT_ATIS_Date (Data Element) -->
<!-- Definition: Date for which a directory entry, or other use, is being requested by a Traveler
or used in a returned message. The precise time at which this date starts may vary with the
time zone of the service. For example: Saturday stay over rates for an airline begin at the
point of departure. -->

<xs:simpleType name="Date">
  <xs:restriction base="xs:date"/>
</xs:simpleType>

<!-- #38 EXT_ATIS_Time (Data Element) -->
<!-- Definition: Time for which a directory entry is being requested by a Traveler or used in a
returned message. -->

<xs:simpleType name="Time">
  <xs:restriction base="xs:time"/>
</xs:simpleType>

<!-- #39 EXT_DmsMessageBeacon (Data Element) -->
<!-- Definition: Indicates if connected beacon(s) are to be activated when the associated message
is displayed. -->

<xs:simpleType name="DmsMessageBeacon" >
  <xs:annotation>
    <xs:appinfo>
      beacons are disabled (0)
      beacons are enabled (1)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="1"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="beacons are disabled"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

```

        <xs:enumeration value="beacons are enabled"/>
    </xs:restriction>
</xs:simpleType >
</xs:union>
</xs:simpleType>

<!-- #40 EXT_DmsMessageMultiString (Data Element) -->
<!-- Definition: Contains the message written in MULTI-language. The value of the MULTI string
is not allowed to have any null character. -->

<xs:complexType name="DmsMessageMultiString" >
    <xs:simpleContent>
        <xs:extension base="DmsMessageMultiString-string" >
            <xs:attribute name="EncodingType" use="required">
                <xs:simpleType>
                    <xs:restriction base="xs:NMTOKEN">
                        <xs:enumeration value="base64Binary"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:attribute>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:simpleType name="DmsMessageMultiString-string">
    <xs:restriction base="xs:base64Binary">
        <xs:minLength value="2"/>
        <xs:maxLength value="342"/>
    </xs:restriction>
</xs:simpleType >

<!-- #41 EXT_DmsMessageNumber (Data Element) -->
<!-- Definition: Enumerated listing of row entries within the value of the primary index to this
table (dmsMessageMemoryType -object). When the primary index is 'currentBuffer' or 'schedule',
then this value must be one (1). When the primary index is 'blank', this value shall be from 1
through 255 and all compliant devices must support all 255 of these 'blank' rows. -->

<xs:simpleType name="DmsMessageNumber" >
    <xs:restriction base="xs:unsignedShort"/>
</xs:simpleType>

<!-- #42 EXT_DmsMessageTimeRemaining (Data Element) -->
<!-- Definition: Indicates the amount of remaining time in minutes that the current message shall
be displayed. The value 65535 indicates an infinite duration. A value of zero (0) shall
indicate that the current message display duration has expired. -->

<xs:simpleType name="DmsMessageTimeRemaining" >
    <xs:restriction base="xs:unsignedShort"/>
</xs:simpleType>

<!-- #43 EXT_DmsSignTechnology (Data Element) -->
<!-- Definition: Indicates the utilized technology in a bitmap format (Hybrids will have to set
the bits for all technologies that the sign utilizes). If a bit is set to one (1), then the
associated feature exists.; if the bit is set to zero (0), then the associated feature does not
exist. -->

<xs:simpleType name="DmsSignTechnology-item" >
    <xs:annotation>
        <xs:appinfo>
            other (0)
            led (1)
            flip disk (2)
            fiber optics (3)
            shuttered (4)
            bulb (5)
            drum (6)
            spare7 (7) -- Adding to make a full byte
        </xs:appinfo>
    </xs:annotation>
    <xs:union>
        <xs:simpleType>

```

```

        <xs:restriction base="xs:int">
            <xs:minInclusive value="0"/>
            <xs:maxInclusive value="7"/>
        </xs:restriction>
    </xs:simpleType>
</xs:simpleType>
<xs:simpleType>
    <xs:restriction base="xs:string">
        <xs:enumeration value="other"/>
        <xs:enumeration value="led"/>
        <xs:enumeration value="flip disk"/>
        <xs:enumeration value="fiber optics"/>
        <xs:enumeration value="shuttered"/>
        <xs:enumeration value="bulb"/>
        <xs:enumeration value="drum"/>
        <xs:enumeration value="spare7"/>
    </xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>
<xs:simpleType name="DmsSignTechnology">
    <xs:list itemType="DmsSignTechnology-item"/>
</xs:simpleType>

<!-- #44 EXT_LRMS_Distance (Data Frame) -->
<!-- Definition: A choice of distance values, in integer or REAL expressions, in different
English and metric units. Note that the data elements for REAL representations are suffixed with
Dec. -->

<xs:complexType name="Distance" >
    <xs:choice>
        <xs:element name="m" >
            <xs:simpleType>
                <xs:restriction base="xs:int">
                    <xs:minInclusive value="-10000000"/>
                    <xs:maxInclusive value="10000000"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="mDec" >
            <xs:simpleType>
                <xs:restriction base="xs:float">
                    <xs:minInclusive value="-10000000"/>
                    <xs:maxInclusive value="10000000"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <!-- Warning, above may need hand editing -->
        <!-- observe that min-max restrictions have -->
        <!-- not been added to this template yet -->
        <xs:element name="mm" >
            <xs:simpleType>
                <xs:restriction base="xs:int">
                    <xs:minInclusive value="-1000000"/>
                    <xs:maxInclusive value="1000000"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="mmDec" >
            <xs:simpleType>
                <xs:restriction base="xs:float">
                    <xs:minInclusive value="-1000000"/>
                    <xs:maxInclusive value="1000000"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <!-- Warning, above may need hand editing -->
        <!-- observe that min-max restrictions have -->
        <!-- not been added to this template yet -->
        <xs:element name="dm" >
            <xs:simpleType>
                <xs:restriction base="xs:int">

```

```

        <xs:minInclusive value="-1000000"/>
        <xs:maxInclusive value="1000000"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="dmDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-1000000"/>
            <xs:maxInclusive value="1000000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="yd" >
    <xs:simpleType>
        <xs:restriction base="xs:int">
            <xs:minInclusive value="-17600000"/>
            <xs:maxInclusive value="17600000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="ydDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-17600000"/>
            <xs:maxInclusive value="17600000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="ft" >
    <xs:simpleType>
        <xs:restriction base="xs:int">
            <xs:minInclusive value="-52800000"/>
            <xs:maxInclusive value="52800000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="ftDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-52800000"/>
            <xs:maxInclusive value="52800000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="in" >
    <xs:simpleType>
        <xs:restriction base="xs:int">
            <xs:minInclusive value="-1000000"/>
            <xs:maxInclusive value="1000000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="inDec" >
    <xs:simpleType>
        <xs:restriction base="xs:float">
            <xs:minInclusive value="-1000000"/>
            <xs:maxInclusive value="1000000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->

```



```

<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="mi" >
  <xs:simpleType>
    <xs:restriction base="xs:short">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="miDec" >
  <xs:simpleType>
    <xs:restriction base="xs:float">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
<xs:element name="km" >
  <xs:simpleType>
    <xs:restriction base="xs:short">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="kmDec" >
  <xs:simpleType>
    <xs:restriction base="xs:float">
      <xs:minInclusive value="-10000"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<!-- Warning, above may need hand editing -->
<!-- observe that min-max restrictions have -->
<!-- not been added to this template yet -->
</xs:choice>
</xs:complexType>
<!-- #45 EXT_LRMS_Height (Data Frame) -->
<!-- Definition: A vertical height expressed as either an altitude with reference to a vertical
datum, or a vertical level from -127 to +127, where level 0 is the ground surface, or ground
level of a structure. -->

<xs:complexType name="Height" >
  <xs:choice>
    <xs:element name="altdatum" >
      <xs:complexType>
        <xs:sequence>
          <xs:element name="altitude" type="Distance" />
          <xs:element name="verticalDatum" type="VerticalDatum" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="verticalLevel" type="VerticalLevel" />
  </xs:choice>
</xs:complexType>

<!-- #46 EXT_LRMS_HorizontalDatum (Data Element) -->
<!-- Definition: The underlying horizontal geodetic datum for a geographic coordinate -->

<xs:simpleType name="HorizontalDatum" >
  <xs:annotation>
    <xs:appinfo>
      wgs 84 (0)
      wgs 84egm 96 (1)
      nad83 (2)
      nad27 (3)
    </xs:appinfo>
  </xs:annotation>
</xs:simpleType>

```

```

        </xs:appinfo>
    </xs:annotation>
    <xs:union>
        <xs:simpleType>
            <xs:restriction base="xs:unsignedInt">
                <xs:minInclusive value="0"/>
                <xs:maxInclusive value="3"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="wgs 84"/>
                <xs:enumeration value="wgs 84egm 96"/>
                <xs:enumeration value="nad83"/>
                <xs:enumeration value="nad27"/>
            </xs:restriction>
        </xs:simpleType >
    </xs:union>
</xs:simpleType>

<!-- #47 EXT_LRMS_Latitude (Data Element) -->
<!-- Definition: The geographic latitude of a node, expressed in integer microdegrees, with
reference to the horizontal datum specified by horizontalDatum. -->

<xs:simpleType name="Latitude" >
    <xs:restriction base="xs:int">
        <xs:minInclusive value="-90000000"/>
        <xs:maxInclusive value="90000000"/>
    </xs:restriction>
</xs:simpleType>

<!-- #48 EXT_LRMS_Longitude (Data Element) -->
<!-- Definition: The geographic longitude of a node, expressed in integer microdegrees, with
reference to the horizontal datum specified by horizontalDatum. -->

<xs:simpleType name="Longitude" >
    <xs:restriction base="xs:int">
        <xs:minInclusive value="-180000000"/>
        <xs:maxInclusive value="180000000"/>
    </xs:restriction>
</xs:simpleType>

<!-- #49 EXT_LRMS_VerticalDatum (Data Element) -->
<!-- Definition: The underlying vertical geodetic datum for a geographic coordinate. -->

<xs:simpleType name="VerticalDatum" >
    <xs:annotation>
        <xs:appinfo>
            wgs 84 (0)
            navd (1)
        </xs:appinfo>
    </xs:annotation>
    <xs:union>
        <xs:simpleType>
            <xs:restriction base="xs:unsignedInt">
                <xs:minInclusive value="0"/>
                <xs:maxInclusive value="1"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="wgs 84"/>
                <xs:enumeration value="navd"/>
            </xs:restriction>
        </xs:simpleType >
    </xs:union>
</xs:simpleType>

<!-- #50 EXT_LRMS_VerticalLevel (Data Element) -->
<!-- Definition: Vertical level specified as ordinal class expressed positive (above) or negative
(below) the ground surface or ground level of a structure (VerticalLevel = 0). -->

```

```

<xs:simpleType name="VerticalLevel" >
  <xs:restriction base="xs:byte">
    <xs:minInclusive value="-127"/>
  </xs:restriction>
</xs:simpleType>

<!-- #51 Filters (Data Frame) -->
<!-- Definition: Filters used in the subscription message process. -->

<xs:complexType name="Filters" >
  <xs:sequence>
    <xs:element name="dms-status-information" >
      <xs:simpleType>
        <xs:restriction base="xs:boolean"/>
      </xs:simpleType>
    </xs:element>
    <xs:element name="ess-status-information" >
      <xs:simpleType>
        <xs:restriction base="xs:boolean"/>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<!-- #52 FreeText (Data Element) -->
<!-- Definition: A simple free-text field used in the message when structured information will
not serve. -->

<xs:simpleType name="FreeText" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="500"/>
  </xs:restriction>
</xs:simpleType>

<!-- #53 InformationalText (Data Element) -->
<!-- Definition: Any set of ASCII characters up to 255. -->

<xs:simpleType name="InformationalText" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="255"/>
  </xs:restriction>
</xs:simpleType>

<!-- #54 LINK_Direction_code (Data Element) -->
<!-- Definition: The direction(s) of travel referenced on a link. -->

<xs:simpleType name="Link-direction" >
  <xs:annotation>
    <xs:appinfo>
      any other (0)
      n (1)
      ne (2)
      e (3)
      se (4)
      s (5)
      sw (6)
      w (7)
      nw (8)
      not directional (9)
      positive direction (10)
      negative direction (11)
      both directions (12)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

```

        <xs:maxInclusive value="12"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="any other"/>
        <xs:enumeration value="n"/>
        <xs:enumeration value="ne"/>
        <xs:enumeration value="e"/>
        <xs:enumeration value="se"/>
        <xs:enumeration value="s"/>
        <xs:enumeration value="sw"/>
        <xs:enumeration value="w"/>
        <xs:enumeration value="nw"/>
        <xs:enumeration value="not directional"/>
        <xs:enumeration value="positive direction"/>
        <xs:enumeration value="negative direction"/>
        <xs:enumeration value="both directions"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

<!-- #55 LINK_RouteDesignator_identifier (Data Element) -->
<!-- Definition: County, State, or Federal route numbers with any associated alphabetic
designators. -->

<xs:simpleType name="Link-route-designator" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>

<!-- #56 ORGANIZATION_CenterIdentifier_identifier (Data Element) -->
<!-- Definition: A unique identification number which identifies an organization's Center
(transformation, emergency management, public safety, etc.) within a region. -->

<xs:simpleType name="Organization-center-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #57 ORGANIZATION_Identifier_identifier (Data Element) -->
<!-- Definition: A unique identifier for an organization within a region. -->

<xs:simpleType name="Organization-identifier" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

<!-- #58 ORGANIZATION_Name_text (Data Element) -->
<!-- Definition: The organization's name. -->

<xs:simpleType name="Organization-name" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>

<!-- #59 ORGANIZATION_SubOrganizationName_text (Data Element) -->
<!-- Definition: The organization's "sub organization" - Department or Bureau, for example -
name. -->

<xs:simpleType name="Organization-sub-organization-name" >
  <xs:restriction base="xs:string">

```

```

        <xs:minLength value="1"/>
        <xs:maxLength value="128"/>
    </xs:restriction>
</xs:simpleType>

<!-- #60 OrganizationInformationLong (Data Frame) -->
<!-- Definition: The long form of the organizational data frame. This is derived from several
TMDD data frames and data elements and has been customized to reflect local DOT needs. The
typical use of this frame is when requesting data about a device, or when building a reply with
data about a device owned by a center. See also the OrganizationInformationShort entry which
has a terse version also derived from the same TMDD data frames and elements. Typically these
are used in pairs. -->

<xs:complexType name="OrganizationInformationLong" >
    <xs:sequence>
        <xs:element name="organization-id" type="Organization-identifier" />
        <xs:element name="organization-name" type="Organization-name" />
        <xs:element name="organization-sub-name" type="Organization-sub-organization-name"
minOccurs="0"/>
        <xs:element name="contactDetails" type="ContactDetails" />
    </xs:sequence>
</xs:complexType>

<!-- #61 OrganizationInformationShort (Data Frame) -->
<!-- Definition: The short form of the organizational data frame. This is derived from several
TMDD data frames and data elements and has been customized to reflect local DOT needs. The
typical use of this frame is when requesting data about a device, or when building a reply with
data about a device owned by a center. See also the OrganizationInformationLong entry which has
a verbose version also derived from the same TMDD data elements. Typically these are used in
pairs. -->

<xs:complexType name="OrganizationInformationShort" >
    <xs:sequence>
        <xs:element name="organization-id" type="Organization-identifier" />
        <xs:element name="organization-name" type="Organization-name" />
    </xs:sequence>
</xs:complexType>

<!-- #62 ReturnAddress (Data Element) -->
<!-- Definition: A URL indicating the subscriber callback message handler. Any set of ASCII
characters up to 128. -->

<xs:simpleType name="ReturnAddress" >
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="128"/>
    </xs:restriction>
</xs:simpleType>

<!-- #63 SECURITY_Password_text (Data Element) -->
<!-- Definition: The unique password of a user to login. -->

<xs:simpleType name="Security-password" >
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="32"/>
    </xs:restriction>
</xs:simpleType>

<!-- #64 SECURITY_UserName_text (Data Element) -->
<!-- Definition: The unique User name used to login to the system. -->

<xs:simpleType name="Security-user-name" >
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="32"/>
    </xs:restriction>
</xs:simpleType>

<!-- #65 SubscriptionAction (Data Frame) -->
<!-- Definition: A set of SubscriptionActionItem entries. -->

```

```

<xs:complexType name="SubscriptionAction" >
  <xs:sequence minOccurs="1" maxOccurs="10">
    <xs:element name="subscriptionAction-item" type="SubscriptionActionItem" />
  </xs:sequence>
</xs:complexType>

<!-- #66 SubscriptionActionItem (Data Element) -->
<!-- Definition: An ENUMERATED list of the subscription actions that a request message may take.
-->

<xs:simpleType name="SubscriptionActionItem" >
  <xs:annotation>
    <xs:appinfo>
      reserved (0)
      newSubscription (1)
      replaceSubscription (2)
      cancelSubscription (3)
      cancelAllPriorSubscriptions (4)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="4"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="newSubscription"/>
        <xs:enumeration value="replaceSubscription"/>
        <xs:enumeration value="cancelSubscription"/>
        <xs:enumeration value="cancelAllPriorSubscriptions"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

<!-- #67 SubscriptionCount (Data Element) -->
<!-- Definition: The nth time the publisher has sent content as part of a description to the
subscriber. -->

<xs:simpleType name="SubscriptionCount" >
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>

<!-- #68 SubscriptionFrequency (Data Element) -->
<!-- Definition: The interval (period) in seconds -->

<xs:simpleType name="SubscriptionFrequency" >
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>

<!-- #69 SubscriptionID (Data Element) -->
<!-- Definition: A unique name created by the subscriber for a subscription. Any set of ASCII
characters up to 128 -->

<xs:simpleType name="SubscriptionID" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>
<!-- #70 SubscriptionName (Data Element) -->

```

```

<!-- Definition: A textual name for the subscription for human readers  NOTE: This is not in the
XML fragment you sent to me. -->

<xs:simpleType name="SubscriptionName" >
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>

<!-- #71 SubscriptionType (Data Frame) -->
<!-- Definition: A collection of SubscriptionTypeItems. -->

<xs:complexType name="SubscriptionType" >
  <xs:sequence minOccurs="1" maxOccurs="10">
    <xs:element name="subscriptionType-item" type="SubscriptionTypeItem" />
  </xs:sequence>
</xs:complexType>

<!-- #72 SubscriptionTypeItem (Data Element) -->
<!-- Definition: An ENUMERATED list of the subscription types that a request message may use. -->

<xs:simpleType name="SubscriptionTypeItem" >
  <xs:annotation>
    <xs:appinfo>
      reserved (0)
      oneTime (1)
      periodic (2)
      onChange (3)
    </xs:appinfo>
  </xs:annotation>
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="3"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="oneTime"/>
        <xs:enumeration value="periodic"/>
        <xs:enumeration value="onChange"/>
      </xs:restriction>
    </xs:simpleType >
  </xs:union>
</xs:simpleType>

<!-- #73 TimeInterval (Data Element) -->
<!-- Definition: A value for how often (in various units form years to seconds) an event should
occur, such as updating a data file. Expressed in the native XML format when used. -->

<xs:simpleType name="TimeInterval">
  <xs:restriction base="xs:duration"/>
</xs:simpleType>

</xs:schema>

```

