Calix Management System (CMS) R14.1 Northbound Interface API Guide

January 2017

#220-00897, Rev 16

Table of Contents

About This Guide

Calix provides open interfaces to Operations Support Systems (OSSs) using industry-standard Transaction Language One (TL1), Simple Network Management Protocol (SNMP), Command Line Interface (CLI), and Extensible Markup Language (XML) protocols. This guide describes the Calix Management System (CMS) XML northbound interface (NBI) for management integration to an OSS.

Service providers can use their OSSs to manage XML-based Calix network elements without significant interface development. This document describes OSS integration for the NBIs on Calix E7 platforms, Calix AE ONTs, Calix C7 networks, and Calix E5-100 Series service units.

Intended Audience

This document assumes users are familiar with Calix systems and related telecommunications equipment, and have had training in system installation and maintenance. The procedures in this guide are of a technical nature and should only be performed by qualified personnel.

Prerequisites

The information and procedures in this guide assume the following:

- Calix equipment and software are properly installed and tested.
- The user has experience and comfort with computer systems, software, and building interfaces using XML documents.
- The user has knowledge of telecommunications and engineering standards.
- Calix network elements (networks, service units, and platforms) must be connected to CMS.
- All templates and profiles required for video, data, and voice service activation have been created and synchronized with the appropriate Calix platforms.

Related Documentation

Access user documentation for all Calix products from the Calix Resource Center online at http://www.calix.com.

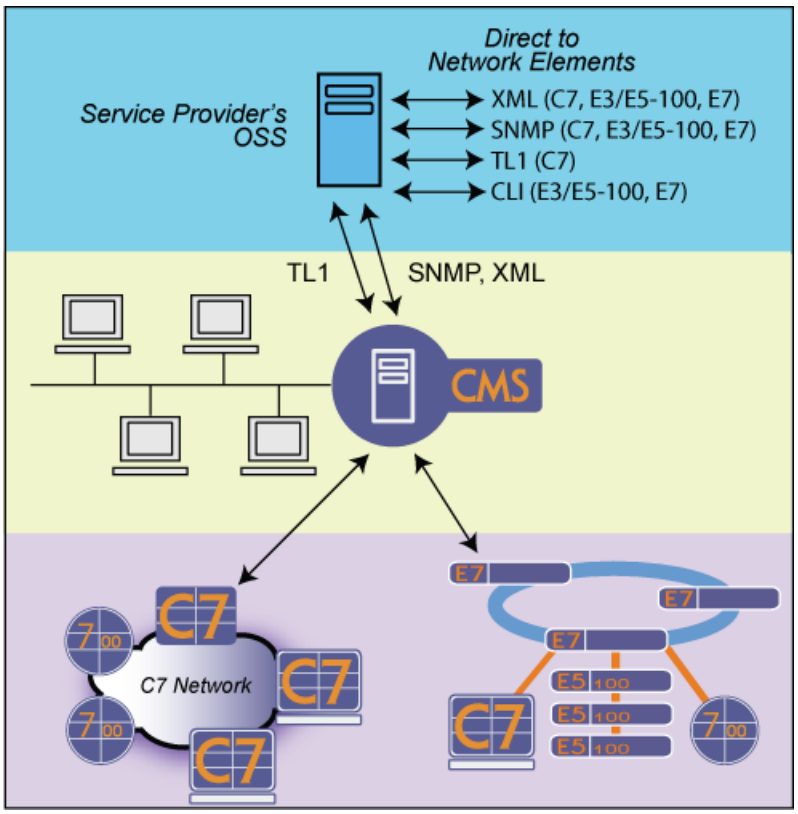
Getting Started

This section describes the following:

Using the XML NBI

The CMS XML NBI provides back-office system integrators with an Application Processing Interface (API) for performing these operations:

- Reading, creating, updating, and deleting subscriber services on Calix C7 line cards, bonding groups, and ONTs; E3/E5-100 xDSL ports, bonding groups, and VoIP ports, E7 GPON ONTs and xDSL ports and bonding groups; and Active Ethernet (AE) ONTs.
- Creating and deleting GPON and AE ONTs.
- Updating subscriber information.
- Manage VLANs on E7 platforms.
- Saving and resetting AE ONTs.
- Suspending and resuming services on Calix C7 ports, bonding groups, and ONTs; E3/E5-100 ports, and E7 GPON ONTs, xDSL ports, and bonding groups.



Each section of this guide lists restrictions that apply to using the XML NBI.

The CMS XML NBI does not support activating subscriber service on the Calix B6 platform.

Logging in to the XML NBI

Guidelines for logging in to the XML NBI

- More than one XML NBI session can be open at the same time. XML NBI login sessions count toward the 150-session client session limit for the CMS server.
- The user name and password must match a CMS user with Full CMS Administration privileges.
- Inactive XML NBI login sessions are governed by the CMS Session Timeout setting (in CMS Desktop, at the CMS level in the Navigation Tree, under **Security** > **Global**), which by default is set to 30 minutes.
- Calix recommends running the login script within the same private network as the CMS server. Accessing a CMS server over a public network is not recommended.

XML NBI Client

Customers must create a client for opening a socket and sending XML requests (posting XML content) over HTTP or HTTPS. The client can be created using a standard programming tool such as Java, Perl, C/C++, PHP, or Python. For an example Java client, including instructions for testing XML requests and replies, contact Calix.

The following table lists the components that must be referenced by the client.

Parameter	Description
host=	IP address or host name of the CMS server.
port=	CMS host server port. For nonsecure (HTTP) login sessions, use 18080. For secure (HTTPS) sessions, use 18443.
URI=	Uniform Resource Indicator used with the host and port to form the URL. Use the following: • For AE ONT requests: **/cmsae/ae/netconf** • For C7/E3/E5-100 requests: **/cmsweb/nc** • For E7 requests: **/cmsexc/ex/netconf**

The following example shows the components in the table for a URL for a nonsecure connection for E7 GPON requests:

http://191.10.20.14:18080/cmsexc/ex/netconf

Sample XML login request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <auth message-id="1">
      <login>
        <UserName>cmsadmin09</UserName>
        <Password>HeadHoncho2</Password>
      </login>
    </auth>
  </soapenv:Body>
</soapenv:Envelope>
```

Parameter	Description
message-id	Identifier for the XML request.
UserName	User name of CMS user account (requires Full CMS Administration privileges)
Password	Password of CMS user account

Sample XML login reply

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <auth-reply xmlns="">
      <ResultCode>0</ResultCode>
      <SessionID>6</SessionID>
    </auth-reply>
  </Body>
</soapenv:Envelope>
```

Element Tag	Description
ResultCode	A successful login returns **0**. If a nonzero integer is returned, a <ResultMessage> tag is also returned with an description of the error.
SessionID	The session ID. This value is required in each XML request.

Logging out of the XML NBI

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <auth message-id="2">
      <logout>
        <UserName>rootgod</UserName>
        <SessionId>6</SessionId>
      </logout>
    </auth>
  </soapenv:Body>
</soapenv:Envelope>
```

XML Operations

For each supported Calix platform, the XML API supports these types of requests:

- **query** or **read**—queries the CMS database for the requested video, data, and voice service provisioning information on a port or ONT and returns the results.
- **create**—activates the requested service(s) on a port or ONT and returns a confirmation of the results.
- **update** or **merge**—appends an existing port or ONT CMS database record and service provisioning on a port or ONT using requested changes and returns a confirmation of the results.
- **delete**—removes the service activation provisioning from a specified port or ONT, as well as from the CMS database, and returns a confirmation of the results.

Card Type Enumerations Table

This matrix shows the CMS database schema ID numbers and the corresponding Calix product name.

Database Value	CMS Web Value		Database Value	CMS Web Value
equip_type	EQUIP-TYPE		actual_type	ACTUAL-TYPE
4	10GE-4		4	10GE-4
5	GPON-4		5	GPON-4
6	GE-12		6	GE-12
7	GE-24		7	GE-24
8	GPON-4x		8	GPON-4x
9	GE-24x		9	GE-24x
10	SCP-10GE		10	SCP-10GE
11	VDSL2-48		11	VDSL2-48
12	VDSL2-48C		12	VDSL2-48C
13	GPON-4R2		13	GPON-4R2
14	GPON-8x		14	GPON-8x
			15	NOT-ALLOWED
16	GPON-8		16	GPON-8
17	E3-48C		17	E3-48C
18	E5-48		18	E5-48
19	VDSL2-48D		19	VDSL2-48D
20	E5-48C		20	E5-48C
21	VDSL2-48r2		21	VDSL2-48r2
22	VDSL2-48Cr2		22	VDSL2-48Cr2
23	VDSL2-48Dr2		23	VDSL2-48Dr2

Database Value	CMS Web Value		Database Value	CMS Web Value
equip_type	EQUIP-TYPE		actual_type	ACTUAL-TYPE
24	E3-8G		24	E3-8G
25	E7-VCP-192		25	E7-VCP-192
26	E7-VCP-384		26	E7-VCP-384
27	E3-48Cr2h		27	E3-48Cr2h
28	E3-48Cr2f		28	E3-48Cr2f
29	E3-48Or2h		29	E3-48Or2h
30	E3-48Or2f		30	E3-48Or2f
31	SCP2-10GE		31	SCP2-10GE
32	GPON-16x		32	GPON-16x

E7 OS xDSL Service Activation and Querying

This section presents the following topics:

Guidelines for E7 OS Provisioning

When using the examples and explanations in this guide, keep in mind the following:

- You must be running CMS server software release 12.0 and Calix E7 OS software release 1.2 or higher.
- In this manual, *E7 OS* denotes the following platforms:
  - E7-2
  - E7-20
  - E3-48C
  - E5-48/C
  - E5-400
- The session ID attribute in the <rpc> tag for E7 OS and AE ONT requests contain a lowercase "i" (sessionid); for other Calix network elements the attribute has an uppercase "I" (sessionId).
- The XML NBI supports all E7 OS services that can be provisioned using the Services screen in CMS Desktop, including pre-provisioned services.
- For service activation for Active Ethernet (AE) ONTs, see "AE ONT Service Activation and Querying" on page 265.
- XML service activation requests override any previously provisioned service parameters.
- For non-required element tags, if no value is provided in the request, do not include the element tag in the XML request.
- Special characters used in the management interface, when returned within element tags, are escaped. Refer to the following table for examples.

Special character	After escaping
<	<
>	>
&	&
"	"

  For example, a global multicast profile name defined as @Video&200 is returned in the XML replies as mcastprof name="@Video&200". Follow company policies and procedures when assigning names in the management interface.
- The E7 OS ONT referenced by the XML attributes must support the requested service. For specific information refer to the Calix E7 OS user documentation.

Common XML Element Tags

This topic covers the following topics:

Note: Element tags for read (get-configuration) E7 OS XML requests are unique. See "XML Read Requests" on page 24.

Common element tags for E7 OS edit-configuration XML Requests

Each E7 OS XML service activation request contains a SOAP envelope and tags, as shown in the following example excerpt.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="37" nodename="NTWK-WestE7" username="JDoe"
        sessionid="14">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              . . .
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML the common element tags in E7 OS **create**, **merge**, and **delete** requests.

Element Tag	Data Type	Req'd ?	Description
`<rpc ...>`	see below	Yes	This tag is nested under <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows.
`message ID=`	Positive Integer: 2^31-1		A unique number identifying the request, used to match the XML reply with the request.

Element Tag	Data Type	Req'd ?	Description
`node name=`	Char(59)	Yes	The case-sensitive name of the E7 OS platform, preceded by **NTWK-** and enclosed in quotes. Example: **"NTWK-Pet02E7"** The nodename value can consist of alphanumeric, underscore, and space characters.
`user name=`	Char	Yes	The name of the user currently logged in to the XML NBI, enclosed in quotes. Example: "**JDoe**"
`session ID=`	Int	Yes	Use the session ID returned after logging in to the CMS NBI (**1** to **150**), enclosed in quotes.
`<edit-config>`	N/A	Yes	This tag is nested under <rpc> tag, and identifies the request as an edit-configuration type.
`<object ...>`	Char	Yes	This tag is nested under the <top> tag, and contains the attributes for defining the XML request, as described in the next two rows.
`operation=`	Char	Yes	Identifies the requested action: • "**create**" • "**merge**" • "**delete**"
`get-config=`	Bool	Yes	Attribute for requesting configuration parameters in the reply: • "**true**"—Include configuration parameters/element tags in the XML reply. • "**false**"—Do not include configuration parameters/element tags in the XML reply.

This section of the document describes specific element tags for each request type nested under the <object> tag.

Common element tags for E7 OS action XML Requests

Action requests are used to apply an xDSL template to a xDSL port or to reset a xDSL port to the factory defaults. For more information, see:

- "Applying an xDSL Template to a Port" on page 56.
- "Resetting an xDSL Port to the Factory Defaults" on page 105

Each E7 OS XML service activation request contains a SOAP envelope and tags, as shown in the following example excerpt.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="37" nodename="NTWK-WestE7" username="JDoe"
        sessionid="14">
      <action>
        . . .
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML the common element tags in E7 OS **action** requests.

Element Tag	Data Type	Req'd ?	Description
`<rpc ...>`	see below	Yes	This tag is nested under <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows.
`message ID=`	Positive Integer: 2^31-1		A unique number identifying the request, used to match the XML reply with the request.
`node name=`	Char(59)	Yes	The case-sensitive name of the E7 OS platform, preceded by **NTWK-** and enclosed in quotes. Example: "**NTWK-Pet02E7**" The nodename value can consist of alphanumeric, underscore, and space characters.
`user name=`	Char	Yes	The name of the user currently logged in to the XML NBI, enclosed in quotes. Example: "**JDoe**"

Element Tag	Data Type	Req'd ?	Description
session ID=	Int	Yes	Use the session ID returned after logging in to the CMS NBI (**1** to **150**), enclosed in quotes.
<action>	N/A	Yes	This tag is nested under <rpc> tag, and identifies the request as an action type.

This section of the document describes specific element tags for each request type nested under the <action> tag.

Common element tags for E7 OS XML replies

Each E7 OS XML reply contains a SOAP envelope and tags, as shown in the following example excerpts.

The following example excerpt shows the results of a successful provisioning request.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="37" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            . . .
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows the results of an unsuccessful provisioning request.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="00359" message-id="100" nodename="NTWK-WestE7">
      <rpc-error>
        <error-type>application</error-type>
        <error-tag>operation-failed</error-tag>
        <error-severity>error</error-severity>
        <error-app-tag>inconsistent-values</error-app-tag>
        <error-message xml:lang="en">tag-action used by ip-host must specify
         expedited COS queue.</error-message>
     </rpc-error>
    </rpc-reply>
   </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML common element tags in E7 replies.

Element Tag	Data Type	Description
`<rpc-reply ...>`	see descr	This tag is nested under <soapenv:Body> tag, and identifies the message ID and E7, as described in the next two rows.
`message-id=`	Positive Integer: 2^31-1	The message ID as provided in the XML request, enclosed in quotes.
`nodename=`	Char(59)	The name of the E7 OS platform, preceded by **NTWK-** and enclosed in quotes. The nodename value can consist of alphanumeric, underscore, and space characters.
Element tag for successful provisioning: `<ok/>`	Tag only	This tag is nested under the <rpc-reply> tag and indicates the request was successful.
Additional element tags for error conditions: `<error-type>` `<error-tag>` `<error-severity>` `<error-app-tag>` `<error-message>`	Char	These tags are nested under the <rpc-reply> tag and indicate the details of the error condition, including the error type, name, severity, application, and message.
`<data>` `<top>` `<object>`	Tag only	For edit-configuration requests, configuration element tags are nested under the <object> tag if the get-config=”true” attribute was included in the XML request.

XML Read Requests

You can retrieve provisioning information using these methods:

- Query VLAN attributes (see "XML get-config requests for VLAN attributes" on page 25)
- Query all services on an xDSL port by service type (see "XML get-config requests for video and data services on a xDSL port" on page 26)
- Query a specific service on an xDSL (see "Query an xDSL port for a service or IP host" on page 31)
- Query an xDSL port by subscriber ID (see "Query an xDSL port by subscriber ID" on page 34)

Common get-config XML element tags

Element Tag	Data Type	Req'd ?	Description
`<rpc ...>`		Yes	This tag is nested under the <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows. The nodename attribute and <ont> element tag values identify the E7 GPON ONT.
`message-ID=`	Positive Integer: 2^31-1		A unique number identifying the request, enclosed in quotes, that is used to match the XML reply with the request.
`nodename=`	Char(59)	Yes	The case-sensitive name of the E7 OS platform, preceded by **NTWK-** and enclosed in quotes. Example: **"NTWK-Pet02E7"** The nodename value can consist of alphanumeric, underscore, and space characters.
`username=`	Char	Yes	The name of the user currently logged in to the CMS XML NBI, enclosed in quotes.

Element Tag	Data Type	Req'd ?	Description
`sessionid=`	Int	Yes	Use the session ID returned after logging in to the CMS XML NBI, enclosed in quotes. **Note:** The session ID attribute for E7 and AE ONT CMS requests contain a lowercase "i" (sessionid); for other Calix network elements the attribute has an uppercase "I" (sessionId).

Use one of the following, depending on the query you are performing (see the query types in this section below):
`<get-config>`
`<show-ont>`
	N/A	Yes	This tag is nested under the <rpc> tag, and identifies the query request type.

`<filter type="subtree">`	see descr	Yes	Identifies the filter or scope of the request. Use the expression to the left in all cases.

XML get-config requests for VLAN attributes

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <get-config>
        <source>
          <running/>
        </source>
        <filter type="subtree">
          <top>
            <object>
              <type>System</type>
              <id/>
              <children>
                <type>Vlan</type>
                <attr-list>name igmp-mode igmp-prof dhcp-snoop mac-force-forw
                    ip-src-verify mac-learn ae-ont-discovery pon-tlan
                    pon-hairpin
                </attr-list>
              </children>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the specific XML element tags used for get-config requests. For element tags common to all get-config requests, see "Common get-config XML element tags" on page 24.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the requested object type. Use the following case-sensitive expression: • **System**
<id> <children> <type>	see descr	Yes	Identifies the requested object. Use the following case-sensitive expression: • **Vlan**
<attr-list>	see descr		Identifies the requested attributes to be returned in the XML reply, which is dependent on the ONT software version. If this tag is not supplied, all attributes are returned.

XML get-config requests for video and data services on a xDSL port

Sample XML request

The following example shows a query for all data and video services on an E7 xDSL port.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="118" nodename="NTWK-WestE7" username="JDoe"
        sessionid="47">
      <get-config>
        <source>
          <running/>
        </source>
        <filter type="subtree">
          <top>
            <object>
              <type>EthIntf</type>
              <id>
                <shelf>9</shelf>
                <card>1</card>
                <ethintf>202</ethintf>
              </id>
              <children>
                <type>EthIntfEthSvc</type>
                <attr-list>admin descr tag-action bw-prof out-tag in-tag
                    mcast-prof</attr-list>
              </children>
```

```
          </object>
        </top>
      </filter>
    </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a query for all SIP services on an E7 xDSL port.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="120" nodename="NTWK-WestE7" username="JDoe"
        sessionid="47">
      <get-config>
        <source>
          <running/>
        </source>
        <filter type="subtree">
          <top>
            <object>
              <type>Pots</type>
              <id>
                <shelf>9</shelf>
                <card>1</card>
                <pots>1</pots>
              </id>
              <children>
                <type>LcSipSvc</type>
                <attr-list>user passwd uri admin sip-prof ip-host call-waiting
                    caller-id-enabled three-way-calling t38-fax-relay
                    </attr-list>
              </children>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the specific XML element tags used for get-config requests for all data and video or VoIP services. See also "Common get-config XML element tags" on page 24.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the provisioning object type using one of the following case-sensitive expressions: • **EthIntf**—data and video service • **Pots**—SIP or TDM gateway VoIP service
Element tags for xDSL voice ports: <id> <shelf> <card> <pots>	see descr	Yes	<shelf> identifies the shelf (**1** to **10**). <card> identifies the card (**1** or **2**). <pots> identifies the port (**1** to **48**).
Element tags for xDSL data/ DSL ports: <id> <shelf> <card> <ethintf>	see descr	Yes	<shelf> identifies the shelf (**1** to **10**) <card> identifies the card (**1** or **2**). <ethintf> identifies the port using a value of 200 + the port number (**201** to **248**).
<children> <type> <attr-list>	see descr	Yes Yes No	<type> identifies the service type object on the E7 xDSL port using one of the following case-sensitive expressions: • **EthIntfEthSvc**—data and video service • **LcSipSvc**—SIP VoIP service • **LcTdmGwSvc**—TDM Gateway VoIP service <attr-list> includes the configuration attributes to return. If no attributes are specified, all attributes are returned. For a list of attributes, refer to the XML examples and element tag tables in the following topics: • "Create data service on E7 OS xDSL ports and xDSL bonding groups" on page 70 • "Create video service on E7 xDSL ports and xDSL bonding groups" on page 78 • "Create SIP VoIP service on E7 OS xDSL ports" on page 91 • "Create TDM Gateway VoIP service on E7 OS xDSL ports" on page 97

Sample XML get-config reply

The following example shows the reply to a query for all data and video services on an E7 xDSL port.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="001074" message-id="118" nodename="NTWK-WestE7">
      <data>
        <top>
          <object>
            <type>EthIntf</type>
            <id>
              <shelf>9</shelf>
              <card>1</card>
              <ethintf>202</ethintf>
            </id>
            <children>
              <child>
                <type>EthIntfEthSvc</type>
                <id>
                  <shelf>9</shelf>
                  <card>1</card>
                  <ethintf>202</ethintf>
                  <ethsvc name="Data1">1</ethsvc>
                </id>
                <admin>enabled</admin>
                <descr />
                <tag-action>
                  <type>SvcTagAction</type>
                  <id>
                    <svctagaction name="@Internet" localId= "4">12
                       </svctagaction>
                  </id>
                </tag-action>
                <bw-prof>
                  <type>BwProf</type>
                  <id>
                    <bwprof name="@Internet_100M" localId"3">12</bwprof>
                  </id>
                </bw-prof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof />
              </child>
            </children>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The attribute name value is returned in the following element tags: <ethsvc name="">, <svctagaction name="">, <bwprof name="">, and <mcastprof name="">, and for the profile element tags, the local profile ID mapped to the global profile is included, as shown in the preceding example.

The following example shows the reply to a query for all SIP voice services on an E7 xDSL port.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="001074" message-id="118" nodename="NTWK-WestE7">
      <data>
        <top>
          <object>
            <type>Pots</type>
            <id>
              <shelf>9</shelf>
              <card>1</card>
              <pots>1</pots>
            </id>
            <children>
              <child>
                <type>LcSipSvc</type>
                  <id>
                    <shelf>9</shelf>
                    <card>1</card>
                    <pots>1</pots>
                    <sipsvc>1</sipsvc>
                  </id>
                  <user>3013032005</user>
                  <passwd>abc123</passwd>
                  <uri>3013032005</uri>
                  <admin>enabled</admin>
                  <sip-prof>
                    <type>SipGwProf</type>
                    <id>
                      <sipgwprof name="@SIP" localId="1">4</sipgwprof>
                    </id>
                  </sip-prof>
                  <ip-host>
                    <type>IpHost</type>
                    <id>
                      <shelf>9</shelf>
                      <card>1</card>
                      <iphost name="SIP">1</iphost>
                    </id>
                  </ip-host>
                  <call-waiting>true</call-waiting>
                  <caller-id-enabled>false</caller-id-enabled>
                  <three-way-calling>true</three-way-calling>
                <t38-fax-relay>false</t38-fax-relay>
              </child>
            </children>
          </object>
```

```
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The attribute name value is returned in the following element tags: <sipgwprofile name="">, and <iphost name="">, and for the profile tag, the local profile ID that the global profile is mapped to is included, as shown in the preceding example.

Query an xDSL port for a service or IP host

Sample XML request

The following example shows a query for data service on E7 xDSL port 2.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="118" nodename="NTWK-WestE7" username="JDoe"
        sessionid="47">
      <get-config>
        <source><running/></source>
        <filter type="subtree">
          <top>
            <object>
              <type>EthIntfEthSvc</type>
              <id>
                <shelf>9</shelf>
                <card>1</card>
                <ethintf>202</ethintf>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the specific XML element tags used for get-config requests for a specific service, port, or object. "Common get-config XML element tags" on page 24.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the service or object using one of the following case-sensitive expressions: • **EthIntfEthSvc**—data or video service • **Pots**—voice ports • **Card**—IP host
For data and video services (with object type EthIntfEthSvc): <id> <shelf> <card> <ethintf> <ethsvc>	see descr	Yes	These element tags identify the port: <shelf> identifies the shelf (**1** to **10**) <card> identifies the card (**1** or **2**). <ethintf> identifies the port using a value of 200 + the port number (**201** to **248**). <ethsvc>—data or video service number (**1** to **12**).
For SIP or TDM Gateway VoIP service (with object type Pots): <id> <shelf> <card> <pots> <children> <type> <attr-list>	see descr	Yes	These element tags identify the port: <shelf> identifies the shelf (**1** to **10**). <card> identifies the card (**1** or **2**). <pots> identifies the voice port (**1** to **48**). <type> identifies the service type (**LcSipSvc** or **LcTdmGwSvc**). <attr-list> includes the configuration attributes to return. If no attributes are specified in the element tag, all attributes are returned. For a list of attributes, refer to the XML examples and element tag tables in the following topics: • "Create SIP VoIP service on E7 OS xDSL ports" on page 91 • "Create TDM Gateway VoIP service on E7 OS xDSL ports" on page 97
For querying the IP host (with object type IpHost): <id> <shelf> <card> <children> <type>	see descr	Yes	These element tags identify the object: <shelf> identifies the shelf (**1** to **10**). <card> identifies the card (**1** or **2**). <type>—identifies the object (**IpHost**).

Sample XML get-config reply

The following example shows a reply to a query for data service on E7 xDSL port 2.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="000545" message-id="147" nodename="NTWK-specialE7″>
      <data>
        <top>
          <object>
            <type>EthIntf</type>
            <id>
              <shelf>9</shelf>
              <card>1</card>
              <ethintf>202</ethintf>
            </id>
            <children>
              <child>
                <type>EthIntfEthSvc</type>
                <admin>enabled</admin>
                <descr/>
                <tag-action>
                  <type>SvcTagAction</type>
                  <id>
                    <svctagaction name=″Test_data_svc″ localId=″24″>3
                        </svctagaction>
                  </id>
                </tag-action>
                <bw-prof>
                  <type>BwProf</type>
                  <id>
                    <bwprof name=″Test_bw_profile″ localId=″12″>1</bwprof>
                  </id>
                </bwprof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof/>
              </child>
            </children>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The attribute name value is returned in the <svctagaction name="">, <bwprof name="">, and <mcastprof name=""> element tags or data and video service, or the <iphost name=""> tag for voice service.

Query an xDSL port by subscriber ID

The following is an example of a query request for filtering the Ethernet interface on xDSL ports by subscriber ID:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="43">
      <get-config>
        <source><running /></source>
        <filter type="subtree">
          <top>
            <object>
              <type>Card</type>
              <id>
                <shelf>9</shelf>
                <card>1</card>
              </id>
              <children>
                <type>EthIntf</type>
                <attr-list>subscr-id</attr-list>
                <attr-filter>
                  <subscr-id>Flinstone</subscr-id>
                </attr-filter>
              </children>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a reply to a query request for filtering the Ethernet interface on xDSL ports by subscriber ID:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="74" nodename="NTWK-E7TRN06">
      <data>
        <top>
         <object>
           <type>Card</type>
             <id>
              <shelf>9</shelf>
              <card>1</card>
             </id>
           <children>
             <child>
               <type>EthIntf</type>
               <id>
                  <shelf>9</shelf>
                  <card>1</card>
                  <ethintf>206</ethintf>
                </id>
                <subscr-id>James</subscr-id>
              </child>
            </children>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Managing VLANs

Creating, Updating, and Deleting VLANs

The XML request defines the objects required to create, read, update, and delete VLANs on the E7 OS.

This section contains the following topics:

For retrieving VLAN attributes, see "XML Read Requests" on page 24.

Create VLANs

Sample XML create request for E7-2

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create" get-config=”true”>
              <type>Vlan</type>
              <id>
                <vlan>22</vlan>
              </id>
              <name>VLAN Test</name>
              <igmp-mode>flood</igmp-mode>
              <igmp-prof>
                <type>VlanIgmpProf</type>
                <id>
                  <vlanigmpprof>1</vlanigmpprof>
                </id>
              </igmp-prof>
              <dhcp-snoop>false</dhcp-snoop>
              <mac-force-forw>true</mac-force-forw>
              <ip-src-verify>false</ip-src-verify>
              <mac-learn>true</mac-learn>
              <ae-ont-discovery>false</ae-ont-discovery>
              <pon-tlan>false</pon-tlan>
              <pon-hairpin>false</pon-hairpin>
            </object>
          </top>
        </config>
```

```
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML create request for E3-48C/E5-48/E5-48C

Note: For E3-48C, E5-48 and E5-48C units, the <shelf> and <card> parameters must be set to 1.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="134" nodename="NTWK-10.245.93.23" timeout="35000"
username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>LcSipSvc</type>
        <id>
         <shelf>1</shelf>
         <card>1</card>
         <pots>48</pots>
         <sipsvc>1</sipsvc>
        </id>
       <user>2121</user>
       <passwd>12121</passwd>
       <uri>1212</uri>
        <sip-prof>
         <type>SipGwProf</type>
        <id>
         <sipgwprof>2</sipgwprof>
        </id>
       </sip-prof>
        <ip-host>
         <type>IpHost</type>
          <id>
           <shelf>1</shelf>
           <card>1</card>
           <iphost>1</iphost>
          </id>
         </ip-host>
          <call-waiting>false</call-waiting>
           <caller-id-enabled>false</caller-id-enabled>
          <three-way-calling>false</three-way-calling>
          <t38-fax-relay>false</t38-fax-relay>
           <dial-plan>
            <type>DialPlan</type>
            <id>
             <dialplan>1</dialplan>
            </id>
```

```
        </dial-plan>
         <enable-msg-waiting-ind>false</enable-msg-waiting-ind>
         <direct-connect/>
        </object>
       </top>
      </config>
     </edit-config>
    </rpc>
   </soapenv:Body>
  </soapenv:Envelope>
```

The following table lists the XML element tags in a **create** request for creating a VLAN on a E7 OS platform. For information about specific VLAN attributes, refer to the Calix E7 OS user documentation. For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <vlan>	See descr	Yes	<type>—Identifies the requested object. Use the following case-sensitive expression: **Vlan** <vlan>—Identifies the VLAN: **2** to **4093**, excluding any reserved VLANs
<name>	Char(31)		Identifies the name of the VLAN. Spaces and special characters are permitted.
<igmp-mode>	See descr		Use one of the following to identify the IGMP mode: • **flood** (default) • **snoop-suppress** • **proxy**
<igmp-prof> <type> <id> <vlanigmpprof>	See descr		<type>—Identifies the profile object using the following case-sensitive expression: **VlanIgmpProfile** <vlanigmpprof>—Numerical identifier for the local profile (**1** to **20**). **Notes:** • The local IGMP profile ID can be viewed in the management interface. In the IGMP profile list, double-click a profile to view its ID above the Name field. • If <igmp-prof> is not specified, the system-default IGMP profile (1) is used.

Element Tag	Data Type	Req'd ?	Description
<dhcp-snoop>	Bool		• **true**—enable DHCP snooping • **false** (default)—disable DHCP snooping
<mac-force-forw>	Bool		• **true**—enable MAC forced forwarding • **false** (default)—disable MAC forced forwarding
<ip-src-verify>	Bool		• **true**—enable IP source verification (binding the IP and MAC addresses to the physical ONT Ethernet port) • **false** (default)—disable IP source verification
<mac-learn>	Bool		(Only applicable for E7-20 and E7-2 standalone systems; Modular Chassis configurations only support MAC learning): • **true** (default)—enable MAC learning • **false**—disable MAC learning
<ae-ont-discovery>	Bool		(Only supported when DHCP snooping is enabled): • **true**—enable AE ONT event discovery • **false** (default)—disable AE ONT event discovery **Note:** This parameter does not apply to E3-48C or E5-48 devices.
<pon-tlan>	Bool		• **true**—enable the PON TLAN parameter • **false** (default)—disable the PON TLAN parameter **Note:** This parameter does not apply to E3-48C or E5-48 devices.
<pon-hairpin>	Bool		(Applicable for TLAN and T1/E3 PWE3 services): • **true**—enable PON hair pining • **false** (default)—disable PON harpooning **Note:** This parameter does not apply to E3-48C or E5-48 devices.

Element Tag	Data Type	Req'd ?	Description
<pppoe-prof> <type> <id> <pppoeprof> **Note:** PPPoE profiles are applicable for E7-2 systems running R2.1 or higher.	See descr		<type>—Identifies the profile using the following case-sensitive expression: **PppoeProfile** <pppoeprof>—Numerical identifier for the local profile. **Note:** The local PPPoE profile ID can be viewed in the management interface. In the PPPoE profile list, double-click a profile to view its ID above the Name field. If not specified, a PPPoE profile is not applied.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
     <soapenv:Body>
        <rpc-reply message-id="2" nodename="NTWK-E7Test">
           <ok/>
           <data>
              <top>
                 <object>
                    <type>Vlan</type>
                    <id>
                       <vlan name="VLAN Test">22</vlan>
                    </id>
                    <name>VLAN Test</name>
                    <mac-learn>true</mac-learn>
                    <igmp-mode>flood</igmp-mode>
                    <dhcp-snoop>false</dhcp-snoop>
                    <mac-force-forw>false</mac-force-forw>
                    <ip-src-verify>false</ip-src-verify>
                    <pon-hairpin>false</pon-hairpin>
                    <pon-tlan>false</pon-tlan>
                    <igmp-prof>
                       <type>VlanIgmpProf</type>
                       <id>
                          <vlanigmpprof name="system-default">1
                              </vlanigmpprof>
                       </id>
                    </igmp-prof>
                    <ae-ont-discovery>false</ae-ont-discovery>
                 </object>
              </top>
           </data>
        </rpc-reply>
     </soapenv:Body>
</soapenv:Envelope>
```

Update VLANs

The element tags that can be included in a merge request are described above, under "Create VLANs"). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

Sample XML merge request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config=”true”>
              <type>Vlan</type>
              <id>
                <vlan>22</vlan>
              </id>
              <pon-tlan>true</pon-tlan>
              <pon-hairpin>true</pon-hairpin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The element tags that can be included in a merge request are described above, under "Create VLANs". If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

Delete VLANs

Sample XML delete request

The following example shows an XML request without the force delete option.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>Vlan</type>
              <id>
                <vlan>100</vlan>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML request with the force delete option.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete" force="true">
              <type>Vlan</type>
              <id>
                <vlan>100</vlan>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The delete VLAN request can be sent with or without a “force” attribute included in the <object> element tag, as described in the following table.

Element Tag and Attributes	Data Type	Req’d?	Description

`<object operation="delete">`
 	
`<object operation="delete" force="true">`
	Force attr is Bool	The force attribute is not required	force=”false”, or force attribute omitted (default)—Do not perform a force delete.
			force=”true”—Perform a force delete (ensures that the VLAN and associated VLAN memberships are deleted).

Adding and Deleting VLAN Members

The XML request defines the objects required to add VLAN members to and delete VLAN members from a VLAN.

This section contains the following topics:

Add a VLAN member

Sample XML create request

The following example shows an XML request to add a LAG interface as a VLAN member.

Note: Before adding a LAG or ERPS domain as a VLAN member, the interface must be created.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>VlanMem</type>
              <id>
                <vlan>333</vlan>
                <vlanmem>0</vlanmem>
              </id>
              <member>
                <type>LagIntf</type>
                <id>
                  <lagintf>1</lagintf>
                </id>
              </member>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML request to add a GE interface as a VLAN member.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="132" nodename="NTWK-e7_20" username="rootgod"
          sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>VlanMem</type>
              <id>
                <vlan>11</vlan>
                <vlanmem>0</vlanmem>
              </id>
              <member>
                <type>EthIntf</type>
                <id>
                  <shelf>1</shelf>
                  <card>1</card>
                  <ethintf>201</ethintf>
                </id>
              </member>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **create** request for adding a VLAN member to a VLAN. For information about specific VLAN attributes, refer to the Calix E7 OS user documentation. For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`<vlan>`
`<vlanmem>`
	See descr	Yes	`<type>`—Identifies the requested object. Use the following case-sensitive expression: **VlanMem**
			`<vlan>`—Identifies the VLAN: 2 to 4093, excluding any reserved VLANs
			`<vlanmem>`—Identifies the VLAN member. Use one of the following:
			• **0**—The E7 auto-generates the VLAN member ID.
			• **1** to **1000**—Supply the VLAN member ID to use (must be unique).

Element Tag	Data Type	Req'd ?	Description
For a LAG interface <member> <type> <id> <lagintf>	See descr	Yes	<type>—Identifies the interface object using the following case-sensitive expression: **LagIntf** <lagintf>—Numerical identifier for LAG interface.
For a GE or 10 GE Ethernet interface <member> <type> <id> <shelf> <card> <ethintf>	See descr	Yes	<type>—Identifies the interface object using the following case-sensitive expression: **EthIntf** <shelf>—Numerical identifier for the shelf. <card>—Numerical identifier for card. <ethintf>—Numerical identifier for GE or 10 GE interface.
For an ERPS domain interface <member> <type> <id> <eapsintf>	See descr	Yes	<type>—Identifies the interface object using the following case-sensitive expression: **EapsIntf** <lagintf>—Numerical identifier for ERPS domain interface.

The VLAN member ID is returned in the XML reply, or you can view VLAN members in the E7 VLAN Provisioning tab in CMS Desktop.

Delete an Interface from a VLAN

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="user14" sessionid="32">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>VlanMem</type>
              <id>
                <vlan>333</vlan>
                <vlanmem>1</vlanmem>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Enabling and Disabling the E7 GE Port Admin Status

Use this command to enable/disable the E7 GE port admin status.

Sample XML Enable Request

```
<rpc message-id="85" nodename="" timeout="35000">
            <edit-config>
                <target>
                    <running/>
                </target>
                <config>
                    <top>
                        <object operation="merge">
                            <type>EthGe</type>
                            <id>
                            <shelf>1</shelf>
                            <card>1</card>
                            <ethge>3</ethge>
                             </id>
                            <admin>enabled</admin>
                            <intf>
                                <type>EthIntf</type>
                                <id>
                                    <shelf>1</shelf>
                                    <card>1</card>
                                    <ethintf>103</ethintf>
                                </id>
                            </intf>
                            <gos>
                                <type>EthPortGos</type>
                                <id>
                                    <ethportgos>1</ethportgos>
                                </id>
                            </gos>
                            <cos>
                                <type>Cos</type>
                                <id>
                                    <cos>1</cos>
                                </id>
                            </cos>
                            <peer-intf/>
                            <bcast-max-rate>off</bcast-max-rate>
                            <unknown-mcast-max-rate>off</unknown-mcast-max-
                               rate>
                            <dlf-max-rate>off</dlf-max-rate>
                            <lacp-prio>32768</lacp-prio>
                            <lacp-timeout>short</lacp-timeout>
                            <prov-duplex>auto</prov-duplex>
                            <flow-control>auto</flow-control>
                            <lldp-mode>enabled</lldp-mode>
                            <speed>auto</speed>
                        </object>
                    </top>
```

```
            </config>
        </edit-config>
    </rpc>
```

Sample XML Disable Request

```
<rpc message-id="85" nodename="" timeout="35000">
        <edit-config>
            <target>
                <running/>
            </target>
            <config>
                <top>
                    <object operation="merge">
                        <type>EthGe</type>
                        <id>
                        <shelf>1</shelf>
                        <card>1</card>
                        <ethge>3</ethge>
                     </id>
                        <admin>disabled</admin>
                        <intf>
                            <type>EthIntf</type>
                            <id>
                                <shelf>1</shelf>
                                <card>1</card>
                                <ethintf>103</ethintf>
                            </id>
                        </intf>
                        <gos>
                            <type>EthPortGos</type>
                            <id>
                                <ethportgos>1</ethportgos>
                            </id>
                        </gos>
                        <cos>
                            <type>Cos</type>
                            <id>
                                <cos>1</cos>
                            </id>
                        </cos>
                        <peer-intf/>
                        <bcast-max-rate>off</bcast-max-rate>
                        <unknown-mcast-max-rate>off</unknown-mcast-max-
                           rate>
                        <dlf-max-rate>off</dlf-max-rate>
                        <lacp-prio>32768</lacp-prio>
                        <lacp-timeout>short</lacp-timeout>
                        <prov-duplex>auto</prov-duplex>
                        <flow-control>auto</flow-control>
                        <lldp-mode>enabled</lldp-mode>
                        <speed>auto</speed>
                    </object>
                </top>
            </config>
```

```
    </edit-config>
</rpc>
```

Provisioning Subscriber Information on E7 OS xDSL Ports

The XML request defines the objects required to add or update subscriber information on an E7 OS xDSL voice or data/DSL port.

Set subscriber information and description for a voice port

The following example shows a **merge** request to set a subscriber ID and description.

Sample XML request to set subscriber ID and description

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body><rpc message-id="294" nodename="" timeout="35000">
    <edit-config>
      <target>
        <running />
      </target>
      <config>
        <top>
          <object operation="merge">
           <type>OntPots</type>
             <id>
               <ont>1</ont>
               <ontslot>6</ontslot>
               <ontpots>2</ontpots>
              </id>
             <admin>enabled</admin>
             <subscr-id>Test hz5</subscr-id>
             <descr>Test hz35</descr>
             <impedance>600-ohm</impedance>
             <signal-type>loop-start</signal-type>
             <system-tx-loss>gr909</system-tx-loss>
             <system-rx-loss>gr909</system-rx-loss>
             <tx-gain-2db>0.0</tx-gain-2db>
             <rx-gain-2db>0.0</rx-gain-2db>
             <nfpa-timer>180</nfpa-timer>
             <nfpa-timer-trig>false</nfpa-timer-trig>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="000089" message-id="294" nodename="" timeout="35000">
    <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Update E7 OS xDSL voice or data port

Sample XML requests to update subscriber information

The following example shows a **merge** request to update subscriber information on an E7 OS xDSL voice port.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="180" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
             <type>Pots</type>
               <id>
                 <shelf>1</shelf>
                 <card>2</card>
                 <pots>12</pots>
               </id>
              <subscr-id>707-123-3456</subscr-id>
              <descr>Bedrock Apts.</descr>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a **merge** request to update subscriber information on an E7 OS xDSL data/DSL port.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="180" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
```

```
          <top>
            <object operation="merge">
             <type>EthIntf</type>
               <id>
                 <shelf>1</shelf>
                 <card>2</card>
                 <ethinf>205</ethinf>
               </id>
              <subscr-id>707-123-3456</subscr-id>
              <desc>Bedrock Apts.</desc>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML update reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="180" nodename="NTWK-e7_20">
      <ok/>
      <data>
        <top>
          <object>
            <type>Pots</type>
            <id>
              <shelf>1</<shelf>
              <card>2</card>
              <pots>12</pots>
            </id>
            <subscr-id>707-123-3456</subscr-id>
            <descr>Bedrock Apts.</descr>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **merge** request for adding or updating the subscriber ID and user description on an E7 OS xDSL port. For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
`<type>`	See descr	Yes	`<type>` identifies the provisioning object type using one of the following case-sensitive expressions: • **Pots**—voice port • **EthIntf**—data/DSL port

Element tags for xDSL voice ports:
`<id>`
`<shelf>`
`<card>`
`<pots>`
	see descr	Yes	`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<pots>` identifies the port (**1** to **48**).

Element tags for xDSL data/DSL ports:
`<id>`
`<shelf>`
`<card>`
`<ethintf>`
	see descr	Yes	`<shelf>` identifies the shelf (**1** to **10**)
			`<card>` identifies the card (**1** or **2**).
			`<ethintf>` identifies the port using a value of 200 + the port number (**201** to **248**).

`<subscr-id>`	Char(27)		Subscriber ID

Element tag for xDSL voice ports:
`<descr>`
 	
Element tag for xDSL data/DSL ports:
`<desc>`
	Char(27)		User description

Provisioning Services on E7 OS xDSL Ports and Bonding Groups

This section covers the following topics:

Applying an xDSL Template to a Port

The XML request defines the objects required to apply an xDSL template to an E7 OS xDSL port before activating service.

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="124" nodename="NTWK-WestE7" username="JDoe"
         sessionid="38">
      <action>
        <action-type>apply-dsl-port-template</action-type>
        <action-args>
          <template>
            <type>DslPortTemplate</type>
            <id>
              <dslporttemplate>21</dslporttemplate>
            </id>
          </template>
          <obj>
            <type>DslPort</type>
            <id>
              <shelf>3</shelf>
              <card>2</card>
              <dslport>1</dslport>
            </id>
          </obj>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **action** request for applying an xDSL template to an E7 OS xDSL port. For descriptions of common XML element tags, see "Common element tags for E7 OS action XML Requests" on page 21.

Element Tag	Data Type	Req'd ?	Description
`<action-type>`	see descr	Yes	`<type>` identifies the provisioning action using the following case-sensitive expression: **apply-dsl-port-template**

`<type>`
`<id>`
`<dslporttemplate>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslPortTemplate**
			`<dslportemplate>` identifies the global xDSL template ID (1 to 256).

Element Tag	Data Type	Req'd ?	Description
<obj> <type> <id> <shelf> <card> <dslport>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **DslPort** <shelf> identifies the shelf on which the card resides (**1** to **10**) <card> identifies the card on which the port resides (**1** or **2**). <dslport> identifies the port to which to apply the template (**1** to **48**).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="124" nodename="NTWK-WestE7"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Managing xDSL Bonding Groups

The XML request defines the objects required for creating and deleting xDSL bonding groups and assigning and removing xDSL ports to and from a bonding group.

Configuration process overview

To activate service on an E7 OS xDSL bonding group, follow these steps:

- Create an xDSL bonding group (see below)
- Assign each member port to the bonding group (requires two XML requests). See "E7 OS xDSL Data Service Activation" on page 70.
  Note: If service has already been activated on the port, it is disabled when the port is assigned to the bonding group interface.
  Note: The XML responses when assigning or removing a port to or from a bonding group are affected by which CMS xDSL bonding group panel is being viewed in the CMS GUI. The response sent when the Bonding Group panel 1 is viewed differs from the response sent when the E7 OS xDSL Port Provisioning/SVC panel is viewed.
- Activate service on the bonding group. See "Create data service on E7 OS xDSL ports and xDSL bonding groups" on page 70 or "Create video service on E7 xDSL ports and xDSL bonding groups" on page 78

To delete an E7 xDSL bonding group, follow these steps:

- Remove the member ports from the bonding group. See "Removing an xDSL port from a bonding group" on page 65.
- Delete the xDSL bonding group. See "Deleting an xDSL bonding group" on page 66.

Creating an xDSL bonding group

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="135" nodename="NTWK-e7_20" username="rootgod"
        sessionid="37">
      <edit-config>
        <target><running/></target>
      <config>
        <top>
          <object operation="create" get-config="true">
            <type>DslBondIntf</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <dslbondintf>0</dslbondintf>
              </id>
              <name>bndname</name>
              <admin>enabled</admin>
              <subscr-id>707-766-1234</subscr-id>
              <desc>customers</desc>
              <igmp-immed-leave>use-vlan-setting</igmp-immed-leave>
              <ds-min-rate>0</ds-min-rate>
              <us-min-rate>0</us-min-rate>
              <sec>
                <type>EthSecProf</type>
                <id>
                  <ethsecprof>1</ethsecprof>
                </id>
              </sec>
              <pbit-map>
                <type>DscpMap</type>
                <id>
                  <dscpmap>1</dscpmap>
                </id>
              </pbit-map>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating an xDSL bonding group. For descriptions of common XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <dslbondintf>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **DslBondIntf** <shelf> identifies the shelf (**1** to **10**). <card> identifies the card (**1** or **2**). <dslbondintf> identifies the bonding group ID using one of these options: • **0**—auto-generate a bonding ID • **1** to **24**—specify a bonding ID
<name>	Char(31)	Yes	Identifies the bonding group name.
<admin>	see descr		Operational status of the bonding interface: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, **enabled** is used.
<subscr-id>	Char(63)		Identifies the subscriber ID.
<desc>	Char(31)		Identifies the description of the bonding group interface.
<igmp-immed-leave>	see descr		Identifies the multicast stream IGMP immediate leave action: • **enabled** • **disabled** • **use-vlan-setting** **Note:** When set this value overrides the IGMP profile setting. If this element tag is not supplied in the XML request, **use-vlan-setting** is used.
<ds-min-rate>	Int		Identifies the minimum upstream rate, in Kbps (**0** to **512000**). When not supplied, **0** is used.
<us-min-rate>	Int		Identifies the minimum downstream rate, in Kbps (**0** to **512000**). When not supplied, **0** is used.

Element Tag	Data Type	Req'd ?	Description
<sec> <type> <id> <ethsecprof>	see descr		<type> identifies the provisioning object as an Ethernet security profile using the following case-sensitive expression: **EthSecProf** <ethsecprof> identifies the global Ethernet Security profile ID (**1** to **16**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **Security** > **Ethernet**). **Note:** If these tags are not supplied, **1** (system-default) is used for the Ethernet Security profile.
<pbit-map> <type> <id> <dscpmap>	see descr		<type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpMap** <dscpmap> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.

Sample XML reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>DslBondIntf</type>
            <id>
              <shelf>1</shelf>
              <card>2</card>
              <dslbondinf name ="bndname">4</dslbondintf>
            </id>
            <admin>enabled</admin>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

If the <object> tag in the request includes the attribute, get-config="true", element tags for the provisioning parameters shown in the above example are returned in the reply.

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Assigning an xDSL port to a bonding group

Note: When assigning a port to a bonding group, the <dslbondintf> tag is used to add the port to the bonding group interface.

Sample XML request

In the following example, port 1 on card 2, shelf 3 is added to bonding group.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="130" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32"
      <edit-config>
        <target><running/></target>
      <config>
        <top>
          <object operation="merge">
            <type>DslPort</type>
              <id>
                <shelf>3</shelf>
                <card>2</card>
                <dslport>1</dslport>
              </id>
              <intf>
                <type>DslBondIntf</type>
                <id>
                  <shelf>3</shelf>
                  <card>2</card>
                  <dslbondintf>1</dslbondintf>
                </id>
              </intf>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for assigning an xDSL port to an E7 xDSL bonding group. For descriptions of common XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <dslport>`
	see descr	Yes	`<type>` identifies the port provisioning object type using the following case-sensitive expression: **DslPort**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslport>` identifies the port to assign (**1** to **48**).

`<intf>`
`  <type>`
`  <id>`
`  <shelf>`
`  <card>`
`  <dslbondintf>`
	see descr	Yes	`<type>` identifies the bonding group provisioning object type using the following case-sensitive expression: **DslBondIntf**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group ID (**1** to **24**).

XML output element tags

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see the preceding table).

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Removing an xDSL port from a bonding group

Note: Assigning a port back to its Ethernet interface <ethintf> removes it from a bonding group.

Sample XML request

In the following example, port 5 on card 2, shelf 1 is assigned to its Ethernet interface, 205.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="130" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32"
      <edit-config>
        <target><running/></target>
      <config>
        <top>
          <object operation="merge">
            <type>DslPort</type>
              <id>
                <shelf>1</shelf>
                <card>2</card>
                <dslport>5</dslport>
              </id>
              <intf>
                <type>EthIntf</type>
                <id>
                  <shelf>1</shelf>
                  <card>2</card>
                  <ethintf>205</ethintf>
                </id>
              </intf>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for assign an xDSL port to its Ethernet interface, removing it from an E7 OS xDSL bonding group. For descriptions of common XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd?	Description
<type> <id> <shelf> <card> <dslport>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **DslPort** <shelf> identifies the shelf (**1** to **10**) <card> identifies the card (**1** or **2**). <dslport> identifies the port to remove (**1** to **48**).
<intf> <type> <id> <shelf> <card> <ethintf>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthIntf** <shelf> identifies the shelf (**1** to **10**) <card> identifies the card (**1** or **2**). <ethintf> identifies the port using a value of 200 + the port number (**201** to **248**).

XML output element tags

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see the preceding table).

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Deleting an xDSL bonding group

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="112" nodename="NTWK-e7_20" username="rootgod"
       sessionid="37">
    <edit-config>
      <target><running/></target>
      <config>
        <top>
```

```
          <object operation="delete" force="true">
            <type>DslBondIntf</type>
            <id>
              <shelf>1</shelf>
              <card>2</card>
              <dslbondintf>4</dslbondintf>
            </id>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
</soapenv:Body></soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **delete** request for deleting an xDSL bonding group. For descriptions of common XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
`<object ...>`	N/A	see descr	In addition to the `operation="delete"` attribute, an optional force attribute can be added: • `force="true"`—force delete the bonding group • `force="false"`—do not force delete (default when attribute is omitted)

`<type>`
`<id>`
`<shelf>`
`<card>`
`<dslbondintf>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslBondIntf**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group ID (**1** to **24**)

Sample XML reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="NTWK-WestE7"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Querying Ports Assigned to a Bonding Group

Use this request to query which ports are assigned to a bonding group.

Sample XML Request

```
<rpc message-id="157" nodename="NTWK-412" timeout="35000">
    <get-config>
        <source>
            <running />
        </source>
        <filter type="subtree">
            <top>
                <object>
                    <type>DslBondIntf</type>
                    <id>
                        <shelf>2</shelf>
                        <card>1</card>
                        <dslbondintf>2</dslbondintf>
                    </id>
                </object>
            </top>
        </filter>
    </get-config>
</rpc>
```

Sample XML Response

```
<rpc-reply message-id="157" nodename="NTWK-412" timeout="35000">
    <data>
        <top>
            <object>
                <type>DslBondIntf</type>
                <id>
                    <shelf>2</shelf>
                    <card>1</card>
                    <dslbondintf name="ttao">2</dslbondintf>
                </id>
                <name>ttao</name>
                <admin>enabled-no-alarms</admin>
                <desc></desc>
                <igmp-immed-leave>use-vlan-setting</igmp-immed-leave>
                <ds-min-rate>0</ds-min-rate>
                <us-min-rate>0</us-min-rate>
                <sec>
                    <type>EthSecProf</type>
                    <id>
                        <ethsecprof name="system-default">1</ethsecprof>
                    </id>
                </sec>
                <pbit-map>
                    <type>DscpMap</type>
                    <id>
```

```
                        <dscpmap name="access">1</dscpmap>
                    </id>
                </pbit-map>
                <subscr-id>1121</subscr-id>
                <force-dot1x>none</force-dot1x>
                <master-port>47</master-port>
            </object>
        </top>
    </data>
</rpc-reply>
```

E7 OS xDSL Data Service Activation

The XML request defines the objects required to create, update, and delete data services on an E7 OS xDSL port or bonding group.

This section contains the following topics:

Create data service on E7 OS xDSL ports and xDSL bonding groups

Sample XML create request

The following example shows a request for data service activation for an xDSL port:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="217" nodename="NTWK-WestE7" username="JDoe"
         sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>EthIntfEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>2</card>
                <ethintf>201</ethintf>
                <ethsvc>1</ethsvc>
              </id>
               <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>10</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
                  <bwprof>34</bwprof>
                </id>
              </bw-prof>
              <out-tag>300</out-tag>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a request for data service activation for an xDSL bonding group:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="217" nodename="NTWK-WestE7" username="JDoe"
         sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>DslBondEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>2</card>
                <dslbondintf>1</dslbondintf>
                <ethsvc>1</ethsvc>
              </id>
               <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>10</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
                  <bwprof>34</bwprof>
                </id>
              </bw-prof>
              <out-tag>300</out-tag>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating data service on an E7 xDSL port or xDSL bonding group. For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description

Element tags for xDSL port service activation:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <ethintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **EthIntfEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**)
			`<card>` identifies the card (**1** or **2**).
			`<ethintf>` identifies the port using a value of 200 + the port number (**201** to **248**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **1** to **8** for data service).

Element tags for xDSL bonding group service activation:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <dslbondintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslBondEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**)
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group (**1** to **24**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **1** to **8** for data service).

`<descr>`	Char(31)		Description of service.
`<admin>`	see descr		Operational status of the xDSL port or bonding group: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, **enabled** is used.

`<tag-action>`
`  <type>`
`  <id>`
`  <svctagaction>`
	see descr	Yes	`<type>` identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction**
			`<svctagaction>` identifies the ID of a pre-defined global service tag action (**1** to **255**).

Element Tag	Data Type	Req'd ?	Description
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (**2** to **4093**, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>EthIntfEthSvc</type>
            <id>
              <shelf>1</shelf>
              <card>2</card>
              <ethinf>201</ethintf>
              <ethsvc>1</ethsvc>
            </id>
            <admin>enabled</admin>
            <descr/>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction name="Test_data_svc" localId="24">10</svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name="Test_bw_profile" localId="12">34</bwprof>
              </id>
            </bwprof>
            <out-tag>300</out-tag>
            <in-tag>none</in-tag>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 72). **Note:** The local IDs of the Service Tag Action and Ethernet Bandwidth profiles that are mapped to the global profile are returned in the XML reply. See the above reply for examples.

Update (merge) data service on E7 OS xDSL ports and bonding groups

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) data service on an E7 OS xDSL port or bonding group is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>EthIntfEthSvc</type>
                . . .
[OR]
              <type>DslBondEthSvc</type>
                . . .
```

Note: The element tags that can be included in a **merge** request for data service on an E7 xDSL port or xDSL bonding group are the same as the input element tags (see "XML input element tags and values" on page 72). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19 and "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 72.

Delete data service on E7 OS xDSL ports and bonding groups

Sample XML delete request

The following example shows a request for deleting service from an xDSL port:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="183" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>EthIntfEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>1</card>
                <ethintf>201</ethintf>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a request for deleting service from an xDSL bonding group:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="183" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>DslBondEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>1</card>
                <dslbondintf>1</dslbondintf>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
```

```
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 19.

The following table lists the XML element tags required in a **delete** data service request for an E7 OS xDSL port or xDSL bonding group.

Element Tag	Data Type	Req'd ?	Description

Element tags for xDSL port service deletion:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <ethintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **EthIntfEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<ethintf>` identifies the port using a value of 200 + the port number (**201** to **248**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **1** to **8** for data service).

Element tags for xDSL bonding group service deletion:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <dslbondintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslBondEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group (**1** to **24**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **1** to **8** for data service).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="183" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** data service request for an E7 OS xDSL port or bonding group include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

E7 OS xDSL Video Service Activation

The XML request defines the objects required to create, update, and delete video services on an E7 OS xDSL port or xDSL bonding group.

This section contains the following topics:

Multicast VLAN Registration (MVR) video service

To activate MVR video service, an MVR profile must already be created and configured on the E7 OS. The MVR profile is referenced by a multicast profile, which is a required element tag in setting up E7 video service.

Create video service on E7 xDSL ports and xDSL bonding groups

Sample XML create request

The following example shows a request for video service activation for an xDSL port:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="226" nodename="NTWK-WestE7" username="JDoe"
        sessionid="83">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>EthIntfEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>1</card>
                <ethintf>201</ethintf>
                <ethsvc>9</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>5</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
```

```
                <bwprof>1</bwprof>
              </id>
            </bwprof>
            <in-tag>10</in-tag>
            <out-tag>101</out-tag>
            <mcast-prof>
              <type>McastProf</type>
              <id>
                <mcastprof>1</mcastprof>
              </id>
            </mcast-prof>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a request for video service activation for an xDSL bonding group:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="226" nodename="NTWK-WestE7" username="JDoe"
        sessionid="83">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>DslBondEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>1</card>
                <dslbondintf>1</dslbondintf>
                <ethsvc>9</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>5</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
                  <bwprof>1</bwprof>
                </id>
              </bwprof>
              <in-tag>10</in-tag>
              <out-tag>101</out-tag>
```

```
            <mcast-prof>
              <type>McastProf</type>
              <id>
                <mcastprof>1</mcastprof>
              </id>
            </mcast-prof>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating video service on an E7 OS xDSL port and bonding group.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description

Element tags for xDSL port service activation:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <ethintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **EthIntfEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<ethintf>` identifies the port using a value of 200 + the port number (**201** to **248**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **9** to **12** for video service).

Element tags for xDSL bonding group service activation:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <dslbondintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslBondEthSvc**
			`<shelf>` identifies the shelf on which the card resides (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group (**1** to **24**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **9** to **12** for video service).

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the port or bonding group: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, **enabled** is used.
<descr>	Char(31)		Description of service.
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (1 to 255).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies an ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id=”226” nodename=”NTWK-WestE7”
      <ok/>
      <data>
        <top>
          <object>
            <type>EthIntfEthSvc</type>
             <id>
              <shelf>1</shelf>
              <card>1</card>
              <ethintf>201</ethintf>
              <ethsvc name=”Video1”>9</ethsvc>
            </id>
            <admin>enabled</admin>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
              <svctagaction name=”Test_data_svc” localId=”3”>1
                   </svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name=”Test_bw_prof” localId=”2”>1</bwprof>
              </id>
            </bwprof>
            <descr/>
            <out-tag>101</out-tag>
            <in-tag>10</in-tag>
            <mcast-prof name=”Test_mcast_prof” localId=”2”>1</mcast-prof>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see “Common element tags for E7 OS XML replies” on page 22.

If the <object> tag in the request includes the attribute, get-config=“true”, element tags for provisioning parameters are returned in the reply (for descriptions, see “XML input element tags and values” on page 80). **Note:** The local IDs of the Service Tag Action, Ethernet bandwidth, and Multicast profiles that are mapped to the global profiles are returned in the XML reply. See the above reply for examples.

Update (merge) video service on E7 OS xDSL ports and xDSL groups

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) video service on an E7 OS xDSL port or xDSL bonding group is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="158" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>EthIntfEthSvc</type>
                . . .
[OR]
              <type>DslBondEthSvc</type>
                . . .
```

Note: The element tags that can be included in a **merge** request for video service on an E7 OS xDSL port or xDSL bonding group are the same as the input element tags (see "XML input element tags and values" on page 80). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19 and "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 80).

Delete video service on E7 OS xDSL ports and xDSL groups

Sample XML delete request

The following example shows a request for deleting service from an xDSL port:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="198" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>EthIntfEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>5</card>
                <ethintf>201</ethintf>
                <ethsvc>9</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows a request for deleting service from an xDSL bonding group:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="198" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>DslBondEthSvc</type>
              <id>
                <shelf>1</shelf>
                <card>5</card>
                <dslbondintf>1</dslbondintf>
                <ethsvc>9</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 19.

The following table lists the XML element tags required in a **delete** video service request on an E7 OS xDSL port or bonding group.

Element Tag	Data Type	Req'd ?	Description

Element tags for xDSL port service deletion:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <ethintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **EthIntfEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**)
			`<card>` identifies the card (**1** or **2**).
			`<ethintf>` identifies the port using a value of 200 + the port number (**201** to **248**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **9** to **12** for video service).

Element tags for xDSL bonding group service deletion:
`<type>`
`<id>`
`  <shelf>`
`  <card>`
`  <dslbondintf>`
`  <ethsvc>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **DslBondEthSvc**
			`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<dslbondintf>` identifies the bonding group (**1** to **24**).
			`<ethsvc>` identifies the data service (**1** to **12**; typically **9** to **12** for video service).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="198" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** video service request for an E7 OS xDSL port or bonding group include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

E7 OS xDSL Voice Service Activation

The XML request defines the objects required to create, update, and delete SIP and TDM Gateway VoIP services on E7 OS xDSL ports.

Before voice service can be activated, you must create an IP host on the E7 OS xDSL card.

This section contains the following topics:

Create an IP host for VoIP service

Sample XML merge requests

The following example creates an IP host for SIP service using DHCP host protocol.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="333" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>IpHost</type>
              <id>
                <shelf>4</shelf>
                <card>2</card>
                <iphost>1</iphost>
              </id>
              <host-proto>dhcp</host-proto>
              <name>iphostname</name>
              <out-tag>18</out-tag>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example creates an IP host for TDM Gateway service with static host protocol.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="334" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>IpHost</type>
              <id>
                <shelf>4</shelf>
                <card>2</card>
                <iphost>1</iphost>
              </id>
              <host-proto>static</host-proto>
              <static-ip>192.11.14.101</static-ip>
              <static-mask>255.255.255.0</static-mask>
              <static-gw>192.11.14.102</static-gw>
              <out-tag>18</out-tag>
              <name>iphostname</name>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for configuring an IP host for voice service on E7 OS xDSL ports.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <iphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **IpHost** <iphost> identifies the IP host ID: **1**

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	Yes No	Outer and inner VLAN IDs (**2** to **4093**, excluding any reserved VLAN IDs). **Note:** If the inner VLAN is not used, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<name>	Char(31)	Yes	Identifies the name of the IP host.
<host-proto>	see descr	Yes	Use one of the following: • **dhcp**—DHCP host protocol • **static**—Static host protocol • **unknown**
Additional element tags for static host protocol: <static-ip> <static-mask> <static-gw>	see descr	Yes	Use the following for these three additional element tags: • static IP address assigned to the ONT, in dotted quad format. • static IP mask assigned to the ONT, in dotted quad format. • static IP gateway for the ONT to use in routing traffic, in dotted quad format (must belong to the same subnet as the subtending IP address) **Note:** For DHCP host protocol, these element tags are not used (though the values are saved in the service record).

XML reply

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="333" nodename="NTWK-WestE7">
      <ok/>
      <data>
        <top>
          <object>
            <type>IpHost</type>
            <id>
              <shelf>3</shelf>
              <card>1</card>
              <iphost name="iphostname">1</iphost>
            </id>
            <out-tag>18</out-tag>
            <in-tag>none</in-tag>
            <host-proto>dhcp</host-proto>
            <static-ip>0.0.0.0</static-ip>
```

```
            <static-mask>0.0.0.0</static-mask>
            <static-gw>0.0.0.0</static-gw>
            <hostname/>
            <name>iphostname</name>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 88).

Create SIP VoIP service on E7 OS xDSL ports

Before SIP voice service can be activated for the first time, you must create an IP host on the E7 OS xDSL card. See "Create an IP host for VoIP service" on page 87.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="240" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>LcSipSvc</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <pots>21</pots>
                <sipsvc>1</sipsvc>
              </id>
              <user>sip_svc</user>
              <passwd>test123</passwd>
              <uri>123</uri>
              <sip-prof>
                <type>SipGwProf</type>
                <id>
                  <sipgwprof>2</sipgwprof>
                </id>
              </sip-prof>
              <admin>enabled</admin>
              <ip-host>
                <type>IpHost</type>
                <id>
                  <shelf>3</shelf>
                  <card>1</card>
                  <iphost>1</iphost>
                </id>
              </ip-host>
              <call-waiting>false</call-waiting>
              <caller-id-enabled>false</caller-id-enabled>
              <three-way-calling>false</three-way-calling>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM Gateway VoIP service on an E7 OS xDSL port.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <pots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcSipSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the voice port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <sipsvc> identifies the voice service (**1**).
<user>	Char	Yes	User name.
<passwd>	Char	Yes	Password for specified user.
<uri>	Char	Yes	Universal resource identifier for SIP service.
<sip-prof> <type> <id> <sipgwprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **SipGwProf** <sipgwprof> identifies the ID of a pre-defined global SIP Gateway profile (**1** to **512**).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.

Element Tag	Data Type	Req'd ?	Description
<ip-host> <type> <shelf> <card> <iphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **IpHost** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the IP host resides on (**1** or **2**). <iphost> identifies the IP host ID: **1**
<call-waiting> <caller-id-enabled> <three-way-calling>	Bool.		These parameters apply to E7 OS platforms operating with E7 R2.1.40 or higher firmware. Valid values are: • **true—**Enabled • **false**—Disabled (default) <call-waiting> Call Waiting feature <caller-id-enabled> Caller ID feature <three-way-calling> Three-Way Calling feature

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="240" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>LcSipSvc</type>
            <id>
              <shelf>3</shelf>
              <card>1</card>
              <pots>21</pots>
              <sipsvc>1</sipsvc>
            </id>
            <admin>enabled</admin>
            <sip-prof>
              <type>SipGwProf</type>
              <id>
                <sipgwprof name="@sip_gw_prof2" localId="4">1</sipgwprof>
              </id>
            </sip-prof>
            <user>sip_usr123</user>
            <passwd>test123</passwd>
            <uri>123</uri>
```

```
            <iphost>
              <type>IpHost</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <iphost name="iphost1">1</iphost>
              </id>
            </iphost>
            <call-waiting>false</call-waiting>
            <caller-id-enabled>false</caller-id-enabled>
            <three-way-calling>false</three-way-calling>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 92). **Note:** The local ID of the SIP gateway profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) SIP voice service on E7 OS xDSL ports

XML merge requests, input element tags, and output element tags

The format for the XML request to update (**merge**) SIP voice service on an E7 OS xDSL port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="230" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>LcSipSvc</type>
                . . .
```

Delete SIP VoIP service on E7 OS xDSL ports

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="806" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>LcSipSvc</type>
              <id>
                <shelf>1</shelf>
                <card>2</card>
                <pots>12</pots>
                <sipsvc>1</sipsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 19.

The following table lists the XML element tags required in a **delete** SIP VoIP service request on an E7 OS xDSL port.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <pots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcSipSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the voice port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <sipsvc> identifies the voice service (**1**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="806" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** SIP VoIP service request for an E7 OS xDSL port include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Create TDM Gateway VoIP service on E7 OS xDSL ports

Before TDM Gateway voice service can be activated for the first time, you must create an IP host on the E7 OS xDSL card. See "Create an IP host for VoIP service" on page 87

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1023" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>LcTdmGwSvc</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <pots>21</pots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
              <tdmgw-prof>
                <type>TdmGwProf</type>
                <id>
                  <tdmgwprof>7</tdmgwprof>
                </id>
              </tdmgw-prof>
              <crv>N1-1-IG1-224</crv>
              <tdmgw-svc-group>
                <type>TDMGWServiceGroup</type>
                <id>
                  <tdmgwservicegroup>2</tdmgwservicegroup>
                </id>
              </tdmgw-svc-group>
              <admin>enabled</admin>
              <ip-host>
                <type>IpHost</type>
                <id>
                  <shelf>3</shelf>
                  <card>1</card>
                  <iphost>1</iphost>
                </id>
              </ip-host>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM Gateway VoIP service on an E7 OS xDSL port.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <pots> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcTdmGwSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <tdmgwsvc> identifies the voice service (**1**).
<tdmgw-prof> <type> <id> <tdmgwprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **TdmGwProf** <tdmgwprof> identifies the ID of a pre-defined global TDM Gateway profile (**1** to **32**).
<crv>	Char	Yes	Call Reference Value for the subscriber line in GR-303 or GR-8 switch interface group. Example: **N1-1-IG1-224** Note the following: • The CRV must be provisioned on the C7 gateway interface group. • The format is case-sensitive, and must be upper case.

Element Tag	Data Type	Req'd ?	Description
<tdmgw-svc-group> <type> <id> <tdmgwservicegroup>	see descr		<type> identifies the object type using the following case-sensitive expression: **TDMGWServiceGroup** <tdmgwservicegroup> identifies the ID of a pre-defined CMS TDM Gateway Service Group (**1** to **1000**). **Note:** If elements tags for the TDM service group are not supplied, the XML reply does not include them.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<ip-host> <type> <shelf> <card> <iphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **IpHost** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the IP host resides on (**1** or **2**). <iphost> identifies the IP host ID: **1**

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1023" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>LcTdmGwSvc</type>
            <id>
              <shelf>3</shelf>
              <card>1</card>
              <pots>21</pots>
              <tdmgwsvc>1</tdmgwsvc>
```

```
            </id>
            <admin>enabled</admin>
            <crv>N1-1-IG1-224</crv>
            <pkt-rate>10ms</pkt-rate>
            <reg-retry-s>3600</reg-retry-s>
            <tdmgw-prof>
              <type=TdmGwProf</type>
              <id>
                <tdmgwprof name="@tdm_gw_prof2" localId="4">2<tdmgwprof>
              </id>
            </tdmgw-prof>
              <tdmgw-svc-group>
                <type>TDMGWServiceGroup</type>
                <id>
                  <tdmgwservicegroup name="svc_grp2">2</tdmgwservicegroup>
                </id>
              </tdmgw-svc-group>
            <iphost>
              <type>IpHost</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <iphost name="iphostname">1</iphost>
              </id>
            </iphost>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 98). **Note:** The local ID of the TDM Gateway profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) TDM Gateway voice service on E7 OS xDSL ports

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) TDM Gateway voice service on an E7 OS xDSL port or bonding group is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="231" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>LcTdmGwSvc</type>
                . . .
```

Delete TDM Gateway VoIP service on E7 OS xDSL ports

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="810" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>LcTdmGwSvc</type>
              <id>
                <shelf>1</shelf>
                <care>2</card>
                <pots>12</pots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 19.

The following table lists the XML element tags required in a **delete** TDM Gateway VoIP service request on an E7 OS xDSL port.

Element Tag	Data Type	Req'd?	Description
<type> <id> <shelf> <card> <pots> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcTdmGwSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <tdmgwsvc> identifies the voice service (**1**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="810" nodename="NTWK-WestE7">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** TDM Gateway VoIP service request for an E7 OS xDSL port include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Suspending and Resuming xDSL Services

The XML request defines the objects required to suspend or resume subscriber service on an E7 OS xDSL voice or data/DSL port.

Update an E7 OS xDSL port

Sample XML request for suspending service

The following shows an example of a suspend (merge) request for an E7 OS xDSL voice port.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="180" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>Pots</type>
              <id>
                <shelf>3</<shelf>
                <card>1</card>
                <pots>24</pots>
              </id>
              <admin>disabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **merge** request for suspending or resuming services on an E7 OS xDSL port. For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
`<type>`	See descr	Yes	`<type>` identifies the provisioning object type using one of the following case-sensitive expressions: • **Pots**—voice port • **DslPort**—data/DSL port

Element Tag	Data Type	Req'd ?	Description

Element tags for xDSL voice ports:
`<id>`
`  <shelf>`
`  <card>`
`  <pots>`
	see descr	Yes	`<shelf>` identifies the shelf (**1** to **10**).
			`<card>` identifies the card (**1** or **2**).
			`<pots>` identifies the port (**1** to **48**).

Element tags for xDSL data/DSL ports:
`<id>`
`  <shelf>`
`  <card>`
`  <dslport>`
	see descr	Yes	`<shelf>` identifies the shelf (**1** to **10**)
			`<card>` identifies the card (**1** or **2**).
			`<dslport>` identifies the port number (**1** to **48**).

`<admin>`	Char		Use one of the following: • **disabled**—suspend all traffic on the port • **enabled**—resume all traffic on the port

Sample XML update reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="2" nodename="NTWK-E7Test">
      <ok/>
      <data>
        <top>
          <object>
            <type>Pots</type>
              <id>
                <shelf>3</shelf>
                <card>1</card>
                <pots>24</pots>
              </id>
            <admin>disabled</admin>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Resetting an xDSL Port to the Factory Defaults

The XML request defines the objects required to reset an E7 OS xDSL port to the factory default settings.

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="124" nodename="NTWK-WestE7" username="JDoe"
         sessionid="38">
      <action>
        <action-type>set-to-default</action-type>
        <action-args>
          <object>
            <type>DslPort</type>
            <id>
              <shelf>1</shelf>
              <card>2</card>
              <dslport>5</dslport>
            </id>
          </object>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in an **action** request for resetting an E7 OS xDSL port to the factory default settings. For descriptions of common XML element tags, see "Common element tags for E7 OS action XML Requests" on page 21.

Element Tag	Data Type	Req'd ?	Description
`<action-type>`	see descr	Yes	`<type>` identifies the provisioning action using the following case-sensitive expression: **set-to-default**
`<object>` `  <type>`	see descr	Yes	`<type>` identifies the port provisioning object type using the following case-sensitive expression: **DslPort**

Element Tag	Data Type	Req'd ?	Description
<id> <shelf> <card> <dslport>	see descr	Yes	<shelf> identifies the shelf on which the card resides (**1** to **10**) <card> identifies the card on which the port resides (**1** or **2**). <dslport> identifies the port to reset (**1** to **48**).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="124" nodename="NTWK-WestE7"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 OS XML replies" on page 22.

Changing the Associated Interface Settings on a Single xDSL Pair

Use this command to change the Security Profile/Associated Interface settings on a single xDSL port.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
     <soapenv:Body>
        <rpc message-id="96" nodename="NTWK-10.245.67.87" timeout="35000"
username="rootgod" sessionid="14">
           <edit-config>
              <target>
                 <running />
              </target>
              <config>
                 <top>
                    <object operation="merge">
                       <type>EthIntf</type>
                       <id>
                          <shelf>1</shelf>
                          <card>1</card>
                          <ethintf>201</ethintf>
                       </id>
                       <role>access</role>
                       <admin>enabled</admin>
                       <desc>dongxiang7</desc>
                       <rstp-act>disabled</rstp-act>
                       <rstp-prio>128</rstp-prio>
                       <rstp-path-cost>8</rstp-path-cost>
                       <policy-map></policy-map>
                       <mtu>1500</mtu>
                       <exp-eth>0x8100</exp-eth>
                       <native-vlan>na</native-vlan>
                       <split-hor>false</split-hor>
                       <bpdu-mac>1d</bpdu-mac>
                       <lacp-tunnel>false</lacp-tunnel>
                       <rstp-edge>false</rstp-edge>
                       <trusted>false</trusted>
                       <bpdu-guard>false</bpdu-guard>
                       <igmp-immed-leave>use-vlan-setting</igmp-immed-
                          leave>
                       <subscr-id>4567</subscr-id>
                       <iqa-mode>no-audit</iqa-mode>
                       <iqa-poll-interval-sec>1</iqa-poll-interval-sec>
                       <iqa-err-per-million-thresh>1000</iqa-err-per-
                           million-thresh>
                       <iqa-poll-window>60</iqa-poll-window>
                       <iqa-interval-cnt-alm-thresh>10</iqa-interval-cnt-
                           alm-thresh>
                       <iqa-min-frame-cnt>100</iqa-min-frame-cnt>
```

```
                        <sec>
                            <type>EthSecProf</type>
                            <id>
                                <ethsecprof>9</ethsecprof>
                            </id>
                        </sec>
                        <pbit-map>
                            <type>DscpMap</type>
                            <id>
                                <dscpmap>1</dscpmap>
                            </id>
                        </pbit-map>
                    </object>
                </top>
            </config>
        </edit-config>
    </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a request for changing the security profile/associated interface settings. For descriptions of common XML element tags, see "Common element tags for E7 OS action XML Requests" on page 21.

Element Tag	Data Type	Req'd ?	Description
`<object>` `  <type>`	see descr	Yes	`<type>` identifies the port provisioning object type using the following case-sensitive expression: **DslPort**

`<id>`
`  <shelf>`
`  <card>`
	see descr	Yes	`<shelf>` identifies the shelf on which the card resides (**1** to **10**)
			`<card>` identifies the card on which the port resides (**1** or **2**).

Provisioning E7 OS SIP Gateway Service

Note: This provisioning applies to E7 OS releases 2.2 and higher. For E7 OS R2.2 and higher, the dial plan needs to be specifically added, otherwise the default value will be used.

Create E7 OS SIP Gateway Service

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
         username="rootgod" sessionid="16">
            <edit-config>
                <target>
                    <running/>
                </target>
                <config>
                    <top>
                        <object operation="create" get-config="true">
                            <type>SipSvc</type>
                            <id>
                                <ont>1</ont>
                                <ontslot>6</ontslot>
                                <ontpots>1</ontpots>
                                <sipsvc>1</sipsvc>
                            </id>
                            <user>test</user>
                            <passwd>test</passwd>
                            <uri>test</uri>
                            <admin>enabled</admin>
                            <sip-prof>
                                <type>SipGwProf</type>
                                <id>
                                    <sipgwprof>2</sipgwprof>
                                </id>
                            </sip-prof>
                            <call-waiting>true</call-waiting>
                            <caller-id-enabled>true</caller-id-enabled>
                            <three-way-calling>true</three-way-calling>
                            <t38-fax-relay>false</t38-fax-relay>
                           <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
                            <dial-plan>
                                <type>DialPlan</type>
                                <id>
                                    <dialplan>1</dialplan>
                                </id>
                            </dial-plan>
                            <enable-msg-waiting-ind>true</enable-msg-waiting-
ind>
                            <direct-connect></direct-connect>
                            <direct-conn-timer-sec>0</direct-conn-timer-sec>
```

```
                    </object>
                </top>
            </config>
        </edit-config>
    </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for E7 OS Gateway SIP service.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <pots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcSipSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the voice port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <sipsvc> identifies the voice service (**1**).
<user>	Char	Yes	User name.
<passwd>	Char	Yes	Password for specified user.
<uri>	Char	Yes	Universal resource identifier for SIP service.
<sip-prof> <type> <id> <sipgwprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **SipGwProf** <sipgwprof> identifies the ID of a pre-defined global SIP Gateway profile (**1** to **512**).

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<ip-host> <type> <shelf> <card> <iphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **IpHost** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the IP host resides on (**1** or **2**). <iphost> identifies the IP host ID: **1**
<call-waiting> <caller-id-enabled> <three-way-calling> <enable-msg-waiting-ind>	Bool.		These parameters apply to E7 OS platforms operating with E7 R2.1.40 or higher firmware. Valid values are: • **true—**Enabled • **false**—Disabled (default) <call-waiting> Call Waiting feature <caller-id-enabled> Caller ID feature <three-way-calling> Three-Way Calling feature <enable-msg-waiting-ind> Message waiting indicator feature
<dial-plan> <type> <id> <dialplan>	see descr		Number plan table for SIP VoIP service. <type> identifies the profile type using the following case-sensitive expression: **DialPlan** <id> The internal ID string for this dial plan. <dialplan> Any established dial plan.

Sample XML Response

```
<soapenv:Envelope>
    <soapenv:Body>
        <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
            <ok/>
            <data>
                <top>
                    <object>
                        <type>SipSvc</type>
                        <id>
                            <ont>1</ont>
                            <ontslot>6</ontslot>
                            <ontpots>1</ontpots>
                            <sipsvc>1</sipsvc>
                        </id>
                        <admin>enabled</admin>
                        <sip-prof>
                            <type>SipGwProf</type>
                            <id>
                                <sipgwprof name="@dd_sip_gw">2</sipgwprof>
                            </id>
                        </sip-prof>
                        <user>test</user>
                        <min>0</min>
                        <warn>0</warn>
                        <passwd>test</passwd>
                        <uri>test</uri>
                        <call-waiting>true</call-waiting>
                        <caller-id-enabled>true</caller-id-enabled>
                        <derived-states>parent-disabled</derived-states>
                        <svc-status/>
                        <call-state/>
                        <dhcp-attempts/>
                        <dhcp-acks/>
                        <dhcp-nacks/>
                        <reg-attempts/>
                        <three-way-calling>true</three-way-calling>
                        <t38-fax-relay>false</t38-fax-relay>
                        <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
                        <dial-plan>
                            <type>DialPlan</type>
                            <id>
                                <dialplan name="system-default" localId="1"/>
                            </id>
                        </dial-plan>
                        <direct-connect/>
                        <direct-conn-timer-sec>0</direct-conn-timer-sec>
                        <enable-msg-waiting-ind>true</enable-msg-waiting-ind>
                        <op-stat>sys-disable</op-stat>
                        <crit>0</crit>
                        <maj>0</maj>
                        <info>0</info>
                        <hook-state/>
                        <config-status/>
```

```
                    <reg-challenges/>
                </object>
            </top>
        </data>
    </rpc-reply>
</soapenv:Body>
</soapenv:Envelope>
```

Delete E7 OS SIP GW Service

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="22">
            <edit-config>
                <target>
                    <running/>
                </target>
                <config>
                    <top>
                        <object operation="delete">
                            <type>SipSvc</type>
                            <id>
                                <ont>1</ont>
                                <ontslot>6</ontslot>
                                <ontpots>1</ontpots>
                                <sipsvc>1</sipsvc>
                            </id>
                        </object>
                    </top>
                </config>
            </edit-config>
        </rpc>
    </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
            <ok/>
        </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

Retrieve E7 OS SIP GW Service

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="22">
            <get>
                <filter type="subtree">
                    <top>
                        <object>
                            <type>SipSvc</type>
                            <id>
                                <ont>1</ont>
                                <ontslot>6</ontslot>
                                <ontpots>1</ontpots>
                                <sipsvc>1</sipsvc>
                            </id>
                        </object>
                    </top>
                </filter>
            </get>
        </rpc>
    </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
 <soapenv:Envelope>
    <soapenv:Body>
        <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
            <data>
                <top>
                    <object>
                        <type>SipSvc</type>
                        <id>
                            <ont>1</ont>
                            <ontslot>6</ontslot>
                            <ontpots>1</ontpots>
                            <sipsvc>1</sipsvc>
                        </id>
                        <op-stat>sys-disable</op-stat>
                        <crit>0</crit>
                        <maj>0</maj>
                        <min>0</min>
                        <warn>0</warn>
                        <info>0</info>
                        <derived-states>parent-disabled</derived-states>
                        <hook-state>unknown</hook-state>
                        <config-status>unknown</config-status>
                        <svc-status>unknown</svc-status>
                        <call-state>unknown</call-state>
                        <dhcp-attempts>0</dhcp-attempts>
                        <dhcp-acks>0</dhcp-acks>
                        <dhcp-nacks>0</dhcp-nacks>
                        <reg-attempts>0</reg-attempts>
```

```
                        <reg-challenges>0</reg-challenges>
                        <reg-rejects>0</reg-rejects>
                        <reg-grants>0</reg-grants>
                        <in-call-attempts>0</in-call-attempts>
                        <in-call-completions>0</in-call-completions>
                        <in-call-busy>0</in-call-busy>
                        <in-call-peer-disconn>0</in-call-peer-disconn>
                        <in-call-ont-disconn>0</in-call-ont-disconn>
                        <out-call-attempts>0</out-call-attempts>
                        <out-call-completions>0</out-call-completions>
                        <out-call-busy>0</out-call-busy>
                        <out-call-peer-disconn>0</out-call-peer-disconn>
                        <out-call-ont-disconn>0</out-call-ont-disconn>
                        <e911-call-attempts>0</e911-call-attempts>
                        <e911-call-completions>0</e911-call-completions>
                        <e911-call-busy>0</e911-call-busy>
                        <e911-call-peer-disconn>0</e911-call-peer-disconn>
                        <e911-call-on-hooks>0</e911-call-on-hooks>
                       <vmwi-notify-msgs-waiting>0</vmwi-notify-msgs-waiting>
                        <vmwi-notify-no-msgs-waiting>0</vmwi-notify-no-msgs-
waiting>
                        <rtp-pkts-sent>0</rtp-pkts-sent>
                        <rtp-pkts-recv>0</rtp-pkts-recv>
                        <rtp-null-ip-sent>0</rtp-null-ip-sent>
                        <rtp-null-ip-recv>0</rtp-null-ip-recv>
                        <act-call-counter>0</act-call-counter>
                        <recv-errors>0</recv-errors>
                        <send-errors>0</send-errors>
                        <encode-type>u-law</encode-type>
                        <rtp-packet-size>unknown</rtp-packet-size>
                        <missing-rtp>0</missing-rtp>
                        <seq-err>0</seq-err>
                        <drop-outs>0</drop-outs>
                        <under-runs>0</under-runs>
                        <listens>0</listens>
                        <recv-comfort-noise-pkts>0</recv-comfort-noise-pkts>
                        <recv-bad-src-ports>0</recv-bad-src-ports>
                        <local-udp-port>0</local-udp-port>
                        <rem-ip-addr>0.0.0.0</rem-ip-addr>
                        <rem-udp-port>0</rem-udp-port>
                        <sec-rem-ip-addr>0.0.0.0</sec-rem-ip-addr>
                        <sec-rem-udp-port>0</sec-rem-udp-port>
                        <qos>0</qos>
                        <ip-line-status>inactive</ip-line-status>
                        <active-911-call>false</active-911-call>
                        <ip-host-mac>00:00:00:00:00:00</ip-host-mac>
                        <ip-host-ip>0.0.0.0</ip-host-ip>
                        <ip-host-mask>0.0.0.0</ip-host-mask>
                        <ip-host-gw>0.0.0.0</ip-host-gw>
                        <supports-all-counters>false</supports-all-counters>
                    </object>
                </top>
            </data>
        </rpc-reply>
    </soapenv:Body>
```

```
</soapenv:Envelope>
```

Update E7 OS SIP GW Service

Sample XML Request

```
 <soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="22">
            <edit-config>
                <target>
                    <running/>
                </target>
                <config>
                    <top>
                        <object operation="merge">
                            <type>SipSvc</type>
                            <id>
                                <ont>1</ont>
                                <ontslot>6</ontslot>
                                <ontpots>1</ontpots>
                                <sipsvc>1</sipsvc>
                            </id>
                            <user>test</user>
                            <passwd>test1</passwd>
                            <uri>test</uri>
                            <admin>enabled</admin>
                            <sip-prof>
                                <type>SipGwProf</type>
                                <id>
                                    <sipgwprof>2</sipgwprof>
                                </id>
                            </sip-prof>
                            <call-waiting>true</call-waiting>
                            <caller-id-enabled>true</caller-id-enabled>
                            <three-way-calling>true</three-way-calling>
                            <t38-fax-relay>false</t38-fax-relay>
                           <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
                            <dial-plan>
                                <type>DialPlan</type>
                                <id>
                                    <dialplan>1</dialplan>
                                </id>
                            </dial-plan>
                           <enable-msg-waiting-ind>true</enable-msg-waiting-
ind>
                            <direct-connect></direct-connect>
                            <direct-conn-timer-sec>0</direct-conn-timer-sec>
                        </object>
                    </top>
                </config>
            </edit-config>
```

```
        </rpc>
    </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
            <ok/>
        </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

Provisioning H.248 Voice Service on E7 OS VDSL

Create H.248 DSL voice service

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="240" nodename="NTWK-412" username="rootgod" sessionid="2">
   <edit-config>
    <target>
    <running />
   </target>
   <config>
    <top>
     <object operation="create" get-config="true">
      <type>LcH248GwSvc</type>
       <id>
        <shelf>2</shelf>
        <card>1</card>
        <pots>8</pots>
        <h248gwsvc>1</h248gwsvc>
       </id>
        <h248gw>
         <type>LcH248Gw</type>
          <id>
           <shelf>2</shelf>
           <card>1</card>
           <h248gw>1</h248gw>
          </id>
         </h248gw>
         <termination-id>8</termination-id>
        </object>
       </top>
      </config>
     </edit-config>
    </rpc>
   </soapenv:Body>
  </soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for E7 OS H.248 DSL voice service.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <shelf> <card> <pots> <h248gwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **LcSipSvc** <shelf> identifies the shelf that the card resides on (**1** to **10**). <card> identifies the card that the voice port resides on (**1** or **2**). <pots> identifies the port number (**1** to **48**). <h248gwsvc> identifies the service type (**1**).
<h248gw> <type> <id> <h248gw>			<type> identifies the profile as an H.248 type using the following case-sensitive expression: **LcH248Gw** <h248gwprof> identifies the ID of a predefined global H.248 profile (1 to 20).
<termination-id>	Char	Yes	Identifies the H.248 termination ID.

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply message-id="240" nodename="NTWK-412">
   <ok/>
   <data>
    <top>
     <object>
      <type>LcH248GwSvc</type>
      <id>
       <shelf>2</shelf>
       <card>1</card>
       <pots>8</pots>
       <h248gwsvc>1</h248gwsvc>
      </id>
       <h248gw>
        <type>LcH248Gw</type>
        <id>
         <shelf>2</shelf>
```

```
        <card>1</card>
        <h248gw name="GW1">1</h248gw>
       </id>
      </h248gw>
      <termination-id>8</termination-id>
      <op-stat>sys-disable</op-stat>
      <crit>0</crit>
      <maj>0</maj>
      <admin>enabled</admin>
      <min>0</min>
      <warn>0</warn>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

E7 GPON ONT Service Activation and Querying

This section presents the following topics:

This chapter refers to provisioning for E7-2 and E7-20 devices only.

Please refer to the chapter "E7 OS xDSL Service Activation and Querying" on page 17 for information about the following topics:

General Reference

- "Common XML Element Tags" on page 123

"Managing VLANs" on page 36

- "Creating, Updating, and Deleting VLANs" on page 36
- "Adding and Deleting VLAN Members" on page 45

Common XML Element Tags

This topic covers the following topics:

- "Common element tags for E7 edit-configuration XML Requests" (see below).
- "XML Read Requests for GPON Service Activation" on page 125.

Common element tags for E7 edit-configuration XML Requests

Each E7 XML service activation request contains a SOAP envelope and tags, as shown in the following example excerpt.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="37" nodename="NTWK-WestE7" username="JDoe"
        sessionid="14">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              . . .
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML the common element tags in E7 **create**, **merge**, and **delete** requests.

Element Tag	Data Type	Req'd ?	Description
<rpc ...>	see below	Yes	This tag is nested under <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows. The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT.
message ID=	Positive Integer: 2^31-1		A unique number identifying the request, used to match the XML reply with the request.

Element Tag	Data Type	Req'd ?	Description
node name=	Char(59)	Yes	The case-sensitive name of the E7 platform, preceded by **NTWK-** and enclosed in quotes. Example: **"NTWK-Pet02E7"** The nodename value can consist of alphanumeric, underscore, and space characters.
user name=	Char	Yes	The name of the user currently logged in to the XML NBI, enclosed in quotes. Example: "**JDoe**"
session ID=	Int	Yes	Use the session ID returned after logging in to the CMS NBI (**1** to **150**), enclosed in quotes.
<edit-config>	N/A	Yes	This tag is nested under <rpc> tag, and identifies the request as an edit-configuration type.
<object ...>	Char	Yes	This tag is nested under the <top> tag, and contains the attributes for defining the XML request, as described in the next two rows.
operation=	Char	Yes	Identifies the requested action: • "**create**" • "**merge**" • "**delete**"
get-config=	Bool	Yes	Attribute for requesting configuration parameters in the reply: • "**true**"—Include configuration parameters/element tags in the XML reply. • "**false**"—Do not include configuration parameters/element tags in the XML reply.

This section of the document describes specific element tags for each request type nested under the <object> tag.

XML Read Requests for GPON Service Activation

You can retrieve provisioning information using these methods:

- Query all services on a GPON ONT by service type (see "Query a GPON ONT for all services by service type" on page 127)
- Query a specific service on a GPON ONT (see "Query a GPON ONT for a specific object or service" on page 130)
- Query E7 GPON ONTs (see "XML show-ont requests for GPON ONTs" on page 135)

Common get-config XML element tags

Element Tag	Data Type	Req'd ?	Description
`<rpc ...>`		Yes	This tag is nested under the <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows. The nodename attribute and <ont> element tag values identify the E7 GPON ONT.
`message-ID=`	Positive Integer: 2^31-1		A unique number identifying the request, enclosed in quotes, that is used to match the XML reply with the request.
`nodename=`	Char(59)	Yes	The case-sensitive name of the E7 platform, preceded by **NTWK-** and enclosed in quotes. Example: **"NTWK-Pet02E7"** The nodename value can consist of alphanumeric, underscore, and space characters.
`username=`	Char	Yes	The name of the user currently logged in to the CMS XML NBI, enclosed in quotes.
`sessionid=`	Int	Yes	Use the session ID returned after logging in to the CMS XML NBI, enclosed in quotes. **Note:** The session ID attribute for E7 and AE ONT CMS requests contain a lowercase "i" (sessionid); for other Calix network elements the attribute has an uppercase "I" (sessionId).

Element Tag	Data Type	Req'd ?	Description

Use one of the following, depending on the query you are performing (see the query types in this section below):
`<get-config>`
`<show-ont>`
	N/A	Yes	This tag is nested under the <rpc> tag, and identifies the query request type.

`<filter type="subtree">`	see descr	Yes	Identifies the filter or scope of the request. Use the expression to the left in all cases.

Query a GPON ONT for all services by service type

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="118" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <get-config>
        <source>
          <running/>
        </source>
        <filter type="subtree">
          <top>
            <object>
              <type>Ont</type>
              <id><ont>766018</ont></id>
              <children>
                <type>EthSvc</type>
                <attr-list>admin descr tag-action bw-prof out-tag
                   in-tag mcast-prof
                </attr-list>
              </children>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the specific XML element tags used for get-config requests for multiple services. See also "Common get-config XML element tags" on page 125.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the provisioning object type using the following case-sensitive expression: **Ont**
<id> <ont>	see descr	Yes	Identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value and <ont> tag value in the <rpc> tag identify the E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<children> <type> <attr-list>	see descr	Yes Yes No	<type> identifies the service type object on the E7 ONT using one of the following case-sensitive expressions: • **EthSvc**—Ethernet data and video service • **H248GwSvc**—H.248 VoIP service • **MgcpGwSvc**—MGCP VoIP service • **Pwe3Svc**—PWE3 DS1 service • **SipSvc**—SIP VoIP service • **TdmGwSvc**—TDM Gateway VoIP service <attr-list> includes the configuration attributes to return. If no attributes are specified in the element tag, all attributes are returned. For a list of attributes, refer to the XML examples and element tag tables in the following topics: • "Create data service on E7 GPON ONTs" on page 167 • "Create video service on E7 GPON ONTs" on page 173 • "There are two types of SIP VoIP service:" on page 188 • "Create TDM Gateway VoIP service on E7 GPON ONTs" on page 198 • "H.248 VoIP service on E7 GPON ONTs" on page 204 • "Create PWE3 service on E7 GPON ONTs" on page 217

Sample XML get-config reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="001074" message-id="118" nodename="NTWK-WestE7">
      <data>
        <top>
          <object>
            <type>Ont</type>
            <id>
              <ont name="">766018</ont>
            </id>
            <children>
              <child>
                <type>EthSvc</type>
                <id>
                  <ont name="">766018</ont>
                  <ontslot>3</ontslot>
                  <ontethany>1</ontethany>
                  <ethsvc name="Data1">1</ethsvc>
                </id>
                <admin>enabled</admin>
                <descr>OAM-EP-1</descr>
                <tag-action>
                  <type>SvcTagAction</type>
                  <id><svctagaction name="DataTag1" localId=”24”>1
                       </svctagaction></id>
                </tag-action>
                <bw-prof>
                  <type>BwProf</type>
                    <id>
                      <bwprof name="af-10M" localId=”6”>4</bwprof>
                    </id>
                  </bw-prof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof></mcast-prof>
              </child>
              <child>
                <type>EthSvc</type>
                <id>
                  <ont name="">766018</ont>
                  <ontslot>3</ontslot>
                  <ontethany>4</ontethany>
                  <ethsvc name="Data1">1</ethsvc>
                </id>
                <admin>enabled</admin>
                <descr></descr>
                <tag-action>
                  <type>SvcTagAction</type>
                  <id>
                  <svctagaction name="CT-Same_UP-MEP" localId=”24”>10
                        </svctagaction>
                  </id>
                </tag-action>
```

```
                <bw-prof>
                  <type>BwProf</type>
                  <id>
                    <bwprof name="be" localId=”6”>10</bwprof>
                  </id>
                </bw-prof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof></mcast-prof>
              </child>
            </children>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

If provisioned, the attribute name value is returned in the following element tags:

- <ont name="">
- Depending on the type of service: <ethsvc name="">, <sipsvc name="">, <tdmgwsvc name="">, <h248gwsvc name="">, or <pwe3svc name="">
- <svctagaction name="">
- <bwprof name="">

Query a GPON ONT for a specific object or service

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id=”118” nodename=”NTWK-WestE7” username=”JDoe”
        sessionid=”37”>
      <get-config>
        <source><running/></source>
        <filter type=”subtree”>
          <top>
            <object>
              <type>EthSvc</type>
              <id>
                <ont>766018</ont>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
```

```
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the specific XML element tags used for get-config requests for a specific service. "Common get-config XML element tags" on page 125.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the service or object using one of the following case-sensitive expressions: Services • **EthSvc**—Ethernet data or video service • **SipSvc**—SIP service • **TdmGwSvc**—TDM Gateway service • **H248GwSvc**—H.248 service • **MgcpGwSvc**—MGCP service • **Pwe3Svc**—PWE3 service RF Video • **OntRfAvo**—RF video IP Host (when a single IP host is queried) • **OntIpHost**—IP host ONT and ONT ports • **Ont**—ONT object, including when multiple IP hosts are queried • **OntDs1**—DS1 ports • **OntEthFe**—FE port • **OntEthGe**—GE port • **OntEthHpna**—HPNA port • **OntPots**—POTS ports

Element Tag	Data Type	Req'd ?	Description

For data and video services (<EthSvc> type):
`<id>`
`  <ont>`
`  <ontslot>`
`  <ontethany>`
`  <ethsvc>`
	see descr	Yes	These element tags identify the port:
			`<ont>`—use the same value as supplied in the `<id><ont>` element tag above.
			`<ontslot>`—port type (one of the following):
			• **3**—Gigabit Ethernet port
			• **4**—HPNA Ethernet port
			• **5**—Fast Ethernet port
			• **8**—RG port.
			• **9**—FB port.
			`<ontethany>`—ONT Ethernet port number (**1** to **8**).
			`<ethsvc>`—data or video service number (**1** to **12**).

For SIP or TDM Gateway VoIP service:
`<id>`
`  <ont>`
`  <ontslot>`
`  <ontpots>`
`  [<sipsvc>|<tdmgwsvc>]`
	see descr	Yes	These element tags identify the port:
			`<ont>`—use the same value as supplied in the `<id><ont>` element tag above.
			`<ontslot>`—ONT voice port (**6**).
			`<ontpots>`—ONT voice port number (**1** to **8**).
			`<sipsvc>` or `<tdmgwsvc>`—voice service number (**1**).

For H.248 or MGCP VoIP service:
`<id>`
`  <ont>`
`  <ontslot>`
`  <ontpots>`
`  [<h248svc>|<mgcpsvc>]`
	see descr	Yes	These element tags identify the port:
			`<ont>`—use the same value as supplied in the `<id><ont>` element tag above.
			`<ontslot>`—ONT voice port (**6**).
			`<ontpots>`—ONT voice port number (**1** to **8**).
			`<h248svc>` or `<mgcpsvc>`—voice service number (**1** to **20**).

Element Tag	Data Type	Req'd ?	Description
For PWE3 service: <id> <ont> <ontslot> <ontds1> <pwe3svc>	see descr	Yes	These element tags identify the port: <ont>—use the same value as supplied in the <id><ont> element tag above. <ontslot>— port type (**7**). <ontds1>—ONT Ethernet port number (**1** to **8**). <pwe3svc>—service number (**1**).
For querying multiple IP hosts: <id> <ont> <children> <type>	see descr	Yes	These element tags identify the object: <ont>—use the same value as supplied in the <id><ont> element tag above. <type>—identifies the object (**OntIpHost**).
For RF video and ONT objects: <id> <ont>	see descr	Yes	<ont>—use the same value as supplied in the <id><ont> element tag above.

Sample XML get-config reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="000545" message-id="147" nodename="NTWK-specialE7″>
      <data>
        <top>
          <object>
            <type>EthSvc</type>
            <id>
              <ont name="">766018</ont>
              <ontslot>3</ontslot>
              <ontethany>1</ontethany>
              <ethsvc name="Data1">1</ethsvc>
            </id>
            <admin>enabled</admin>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction name=″Test_data_svc″ localId=″24″>3
                    </svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name=″Test_bw_profile″ localId=″12″>1</bwprof>
              </id>
            </bwprof>
            <descr/>
            <out-tag>none</out-tag>
            <in-tag>none</in-tag>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

If provisioned, the attribute name value is returned in the following element tags:

- <ont name="">
- Depending on the service type: <ethsvc name="">, <sipsvc name="">, <tdmgwsvc name="">, <h248gwsvc name="">, or <pwe3svc name="">
- <svctagaction name="">, <bwprof name="">, and <mcastprof name="">

XML show-ont requests for GPON ONTs

The following is an example of a show-ont query request for filtering ONTs by subscriber ID:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="33">
      <action>
        <action-type>show-ont</action-type>
        <action-args>
          <subscr-id>707-766-3500/subscri-id>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a show-ont query request for filtering ONTs by serial number:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="33">
      <action>
        <action-type>show-ont</action-type>
        <action-args>
          <serno>79c25</serno>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a show-ont query request for filtering ONTs by registration number:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="175" nodename="NTWK-e7_20" username="rootgod"
        sessionid="33">
      <action>
        <action-type>show-ont</action-type>
        <action-args>
          <reg-id>7775554444</reg-id>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags used for get requests for ONT statistics. For element tags common to all get-config requests, see "Common get-config XML element tags" on page 125.

Element Tag	Data Type	Req'd ?	Description
`<action>` `  <action-type>`	see descr	Yes	Identifies the requested action type. Use the following case-sensitive expression: • **show-ont**

Use one of the following methods to query an ONT:
 	
`<action-args>`
`  <subscr-id>`
 	
`<action-args>`
`  <serno>`
 	
`<action-args>`
`  <reg-id>`
	see descr	Yes	**Note:** The match must be exact. Partial matches of the attributes are not supported.
			`<subscr-id>` ONT subscriber ID that matches back-office ID.
			`<serno>` Hexadecimal representation of the ONT serial number.
			`<reg-id>` ONT registration ID that is the RONTA identifier.

Sample XML read reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="000774" message-id="52"
    <ok/>
      <action-reply>
        <match>
          <get-config>
            <object>
              <type>Ont</type>
              <id>
                <ont>1</ont>
              </id>
              <admin>enabled</admin>
              <ontprof>
              <type>OntProf</type>
              <id>
                <ontprof name="767GX-R">137</ontprof>
              </id>
            </ontprof>
            <serno>1234</serno>
            <reg-id></reg-id>
            <subscr-id>707-766-3500</subscr-id>
            <descr>123 Main St.</descr>
            <linked-pon></linked-pon>
```

```
            <pwe3prof><type>OntPwe3Prof</type>
              <id>
               <ontpwe3prof name="system-default" localId=”10”>1</ontpwe3prof>
              </id>
            </pwe3prof>
          </object>
          </get-config>
          <get>
            <object>
              <type>Ont</type>
              <id>
                <ont>1</ont>
              </id>
              <op-stat>sys-disable</op-stat>
              <crit>0</crit>
              <maj>0</maj>
              <min>0</min>
              <warn>0</warn>
              <info>0</info>
              <derived-states>child-prov</derived-states>
              <model></model>
              <vendor></vendor>
              <clei></clei>
            </object>
          </get>
        </match>
      </action-reply>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Adding or Updating Subscriber Information on E7 GPON ONT Ports

The XML request defines the objects required to add or update subscriber information on an E7 ONT Ethernet, Voice, DS1, RF video, or Hot RF video port.

Update E7 ONT port

Sample XML request to update subscriber information

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="180" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntEthGe</type>
              <id>
                <ont>3</<ont>
                <ontslot>3</ontslot>
                <ontethge>1</onethge>
              </id>
              <subscr-id>707-123-2345</subscr-id>
              <descr>123 Main St</descr>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML update reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="2" nodename="NTWK-E7Test">
      <ok/>
      <data>
        <top>
          <object>
```

```
            <type>OntEthGe</type>
            <id>
              <ont>3</<ont>
              <ontslot>3</ontslot>
              <ontethge>1</onethge>
            </id>
            <subscr-id>707-123-2345</subscr-id>
            <descr>Bedrock Apts.</descr>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **merge** request for adding or updating the subscriber ID and user description on an E7 GPON port. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
`<type>`	See descr	Yes	`<type>` identifies the provisioning object type using one of the following case-sensitive expressions: • **OntDs1**—DS1 ports • **OntEthFe**—FE port • **OntEthGe**—GE port • **OntEthHpna**—HPNA port • **OntPots**—POTS ports • **OntRfAvo**—RF video

`<id>`
`  <ont>`
`  <ontslot>`
`  [<ontethge>|`
`  <ontethhpna>|`
`  <ontethfe>|`
`  <ontpots>|`
`  <ontds1>]`
	see descr	Yes	These element tags identify the port:
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—port type (one of the following):
			• **1**—RF Video
			• **2**—Hot RF Video
			• **3**—Gigabit Ethernet port
			• **4**—HPNA Ethernet port
			• **5**—Fast Ethernet port
			• **6**—Voice port
			• **7**—PWE3 port
			[<ontethge>|<ontethhpna>|<ontethfe>|<ontpots>|<ontds1>|<ontvideorf>|<ontvideohotrf>]—port number

Element Tag	Data Type	Req'd ?	Description
<subscr-id>	Char(27)		Subscriber ID
<descr>	Char(27)		User description

Provisioning Services on E7 GPON ONTs

Creating and Deleting E7 GPON ONTs

The XML request defines the objects required to create and delete E7 GPON ONTs.

This section contains the following topics:

E7 GPON ONT Profile IDs

The following profile IDs are used when creating E7 GPON ONTs:

Model	Profile ID
710	101
710G	102
710GX	103
711	104
711G	105
711GE	138
711GX	106
712	107
712G	108
712GE	149
712GX	109
714	110
714G	111
714GX	112
716GE	139
717GE	140
720	113
720G	114
720GX	115
721	116
721G	117
721GE	141
721GX	**118**

Model	Profile ID
722	119
722G	120
722GE	150
722GX	121
724	122
724G	123
724GX	124
725	125
725G	126
725GE	144
725GX	127
726GE	142
727GE	143
742GE	145
743GE	146
744GE	147
760G	128
760GX	129
762G	130
763GX	132
763GX-R	134
766GX	133
766GX-R	135

Model	Profile ID
766GX-R-24	136
767GX-R	137
801G	161
803G	164
812G	162
813G	163
836GE	148
844G	157
844GE	165
854G	158
GIA	156
T071G	151
T072G	152
T073G	153
T076G	154
T077G	155
T710G	159
T720G	160

Create E7 GPON ONTs

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="45" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>Ont</type>
              <id>
                <ont>2</ont>
              </id>
              <admin>enabled</admin>
              <ontprof>
                <type>OntProf</type>
                <id>
                  <ontprof>143</ontprof>
                </id>
              </ontprof>
              <serno>0</serno>
              <reg-id>7775552222</reg-id>
              <subscr-id>707-766-3500</subscr-id>
              <descr>123 Main St</descr>
              <pwe3prof>
                <type>OntPwe3Prof</type>
                <id>
                  <ontpwe3prof>1</ontpwe3prof>
                </id>
              </pwe3prof>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a create request for creating an ONT on a E7 platform. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <ont> identifies the ONT by its E7 scope ID. Use one of the following values: • **0** to have the system automatically assign the next available ON T ID number. • 1 to 64000000 to assign a specific ONT ID. **Note:** The nodename attribute value in the <rpc> tag and <ont> element tag values identify the GPON ONT.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<ontprof> <type> <id> <ontprof>	see descr	Yes	<type> identifies the profile type as ONT using the following case-sensitive expression: **OntProf** <ontprof> identifies the ID of a pre-defined local ONT profile, which can be a custom profile from **1** to **50**, or one of the default profile IDs listed in "Create E7 GPON ONTs" on page 143.
<serno>	Hex(8)		Enter one of the following: • **0**—Leave the serial field empty. • Hexadecimal representation of the ONT serial number.
<reg-id>	Int(10)		ONT registration ID that is the RONTA identifier.
<subscr-id>	Char		ONT subscriber ID that matches back-office ID.

Element Tag	Data Type	Req'd ?	Description
<descr>	Char(48)		Description that labels the ONT location.
<pwe3prof> <type> <id> <ontpwe3prof>	see descr		<type> identifies the profile type as PWE3 using the following case-sensitive expression: **OntPwe3Prof** <ontpwe3prof> identifies the ID of the profile that sets the ONT PWE3 mode. Use **1** (also the default, if not supplied) for the system-default profile, which is set to use either T1 or E1 mode in the management interface.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id=”45” nodename=”NTWK-WestE7”
      <ok/>
      <data>
        <top>
          <object>
            <type>Ont</type>
             <id>
              <ont name=””>2</ont>
            </id>
            <admin>enabled</admin>
            <ontprof>
              <type>OntProf</type>
              <id>
                <ontprof name=”727GE”>143</ontprof>
              </id>
            </ontprof>
            <serno>0</serno>
            <reg-id>7775552222</reg-id>
            <subscr-id>Test_5</subscr-id>
            <descr>123 Main St</descr>
            <linked-pon/>
            <pwe3prof>
              <type>OntPwe3Prof</type>
              <id><ontpwe3prof name=”system-default” localId=”10”>1
                   </ontpwe3prof></id>
            </pwe3prof>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for E7 edit-configuration XML Requests" on page 123.

Note the following.

Element Tag	Data Type	Description

`<ontprof>`
`<type>`
`<id>`
`<ontprof>`
	see descr	`<type>` identifies the profile type as ONT using the following case-sensitive expression: **OntProf**
		
		`<ontprof>` identifies the ID of a pre-defined local ONT profile, which can be a custom profile (**1** to **50**), or one of the default profile IDs listed in "Create E7 GPON ONTs" on page 143.

If the <object> tag includes the attribute get-config="true", the element tags with the provisioning parameters are returned (for descriptions, see "XML input element tags and values" on page 144).

Delete E7 GPON ONTs

Sample XML delete request

The following examples shows a ONT delete request without the force option.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="23" nodename="NTWK-WestE7" username="JDoe"
        sessionid="11">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>Ont</type>
              <id>
                <ont>2</ont>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following examples shows a ONT delete request with the force option.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="23" nodename="NTWK-WestE7" username="JDoe"
        sessionid="11">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete" force="true">
              <type>Ont</type>
              <id>
                <ont>2</ont>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** request for an E7 GPON ONT.

Element Tag	Data Type	Req'd?	Description

`<object operation="delete">`
 	
`<object operation="delete" force="true">`
	Force attribute is Bool	The force attribute is not required	force="false", or force attribute omitted (default)—Do not perform a force delete. **Note:** For a non-force delete to be successful, all service must be removed from the ONT.
			force="true"—Perform a force delete (deletes that all services on the ONT).

`<type>`
`<id>`
`  <ont>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> element tag value identify the GPON ONT.

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="23" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for an E7 ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common get-config XML element tags" on page 125.

Configuring a GPON ONT Ethernet Port

The XML request defines the objects required to configure Ethernet port settings on an E7 GPON ONT.

The following section contains the following examples:

Provision a GE port as an RG or FB interface

Sample XML request for provisioning a GE port as an RG interface

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="137" nodename="NTWK-412" timeout="35000" username="rootgod"
   sessionid="62">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>OntEthGe</type>
       <id>
        <ont>1</ont>
        <ontslot>3</ontslot>
        <ontethge>1</ontethge>
       </id>
       <intf>
        <type>OntRg</type>
        <id>
         <ont>1</ont>
         <ontslot>9</ontslot>
         <ontrg>1</ontrg>
        </id>
       </intf>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for provisioning a GE port as an RG or HB interface.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntEthGe**—GE port
<id> <ont> <ontslot> <ontethge>	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—port type: • **3**—Gigabit Ethernet port <ontethge>—port number
<intf> <type> <id> <ont> <ontslot> <ontrg>	see desc	Yes	<type>—identifies the provisioning object type using one of the following case-sensitive expressions: — **OntRg** - this expression in the example specifies an ONT RG port — **OntFb** - This expression would specify an ONT FB port <ont>—identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—identifies the port type • the number **9** identifies the port type as an FB port. • the number **8** in the example would identify the port type as an RG port. <ontrg>—Identifies the RG port number. If an FB port was being configured, this element would be <ontfb>

Provision an RG interface with Native Mgmt Mode and WAN Protocol DHCP

This section describes how to provision an RG interface with the Management mode configured as native and the WAN protocol configured for DHCP. Native is the default behavior where the CMS/E7 directly provisions the RG WAN interface.

Please refer to "Provision an RG interface with Native Mgmt Mode and WAN Protocol PPPoE" on page 155 for information about how to configure the WAN protocol for PPPoE.

Note: T-Series ONTs do not support Native Mode as an option for the Management Mode.

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
     <soapenv:Body>
        <rpc message-id="43" nodename="AeCMSNetwork" username="rootgod"
sessionid="13">
           <edit-config>
              <target>
                 <running/>
              </target>
              <config>
                 <top>
                    <object operation="merge">
                       <type>OntRg</type>
                       <id>
                          <ont>2</ont>
                          <ontslot>8</ontslot>
                          <ontrg>1</ontrg>
                       </id>
                       <admin>enabled-no-alarms</admin>
                       <subscr-id/>
                       <descr/>
                       <mgmt-mode>native<</mgmt-mode>
                       <wan-protocol>dhcp</wan-protocol>
                       <static-ip>0.0.0.0</static-ip>
                       <static-ip-mask>0.0.0.0</static-ip-mask>
                       <static-ip-gw>0.0.0.0</static-ip-gw>
                       <pri-dns-server>0.0.0.0</pri-dns-server>
                       <sec-dns-server>0.0.0.0</sec-dns-server>
                       <pppoe-user/>
                       <mgmt-prof>
                          <type>OntRgMgmtProf</type>
                          <id>
                             <ontrgmgmtprof>1</ontrgmgmtprof>
                          </id>
                       </mgmt-prof>
                       <tr69-eth-svc>
                          <type>EthSvc</type>
                          <id>
                             <ont>2</ont>
                             <ontslot>8</ontslot>
```

```
                                <ontethany>1</ontethany>
                                <ethsvc>1</ethsvc>
                            </id>
                        </tr69-eth-svc>
                        <default-wan-svc>
                            <type>EthSvc</type>
                            <id>
                                <ont>2</ont>
                                <ontslot>8</ontslot>
                                <ontethany>1</ontethany>
                                <ethsvc>1</ethsvc>
                            </id>
                        </default-wan-svc>
                        <disable-on-batt>true</disable-on-batt>
                        <ont>CXNK00205222</ont>
                    </object>
                </top>
            </config>
        </edit-config>
    </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

Sample XML response for provisioning an RG interface (Native mode and DHCP)

```
<soapenv:Envelope>
    <soapenv:Body>
        <rpc-reply message-id="43" nodename="AeCMSNetwork">
            <ok/>
        </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for provisioning an RG interface (Native mode and DHCP).

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the Remote Gateway (FB) port is the provisioning object type using the following case-sensitive expression - **OntRg**
<id> <ont> <ontslot> <ontrg>	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—The number **8** specifies an RG port <ontrg>—RG port number
<admin>	see descr		Configure the admin state as **enabled** to enable all traffic on the port
<subscr-id>	Char(63		Subscriber ID
<descr>	Char(31)		User description
<mgmt-mode>	see descr		The management mode is configured with the case sensitive word **native**. This is the default value when a new ONT is created. Native management mode allows the service provisioning from the E7. The Static and PPPoE parameters are applicable only in this case. The WAN protocol (dhcp, static, pppoe) is applicable only in the native management mode. ---------- The other available management mode is **external**. External management mode allows the RF service provisioning by an external RF configuration file or ACS. **Note:** T-Series ONTs do not support Native Mode as an option for the Management Mode.

Element Tag	Data Type	Req'd ?	Description
<wan-protocol>	see descr		The WAN protocol is configured with the case sensitive word **dhcp.** The other options are **static** and **pppoe**.
<mgmt-prof> <type> <id> <ontrgmgmtprof>	see descr		<type>- identifies the RG Management profile with the case sensitive word **OntRgMgmtProf** <ontrgmgmtprof>- identifies the RG Management profile number
<tr69-eth-svc> <type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr		<type> identifies the TR-69 Eth Service with the case sensitive word **EthSvc**. <ont>—identifies the ONT by its E7 scope ID (1 to 64000000) <ontslot>—The number **8** identifies the port type as an RG port. <ontethany>- identifies the ONT port number <ethsvc> - identifies the data service (1 to 12; typically 1 to 8 for data service).
<tr69-out-tag> <tr69-in-tag>	see descr		Configure the outer and inner tag VLAN ID tags for the access channel (**none**, **1** to **4093).** The content of these tags depends on the management profile used by the RG interface.
<disable-on-batt>	Bool		Identifies the port operational state when the ONT is operating on battery backup power: • **false**—No (default if no tag is provided) • **true—Yes**
<pbit-map> <type> <id> <dscpmap>	see descr		<type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpMap** <dscpmap> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.

Provision an RG interface with Native Mgmt Mode and WAN Protocol PPPoE

This section describes how to provision an RG interface with the Mgmt mode configured as native and the WAN protocol configured as PPPoE.

Please refer to "Provision an RG interface with Native Mgmt Mode and WAN Protocol DHCP" on page 151 for information about how to configure the WAN protocol for DHCP.

Note: T-Series ONTs do not support Native Mode as an option for the Management Mode.

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="137" nodename="NTWK-412" timeout="35000"
   username="rootgod" sessionid="62">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>OntRg</type>
       <id>
        <ont>1</ont>
        <ontslot>8</ontslot>
        <ontrg>1</ontrg>
       </id>
       <admin>enabled</admin>
       <subscr-id>dan</subscr-id>
       <descr>daniel</descr>
       <mgmt-mode>native</mgmt-mode>
       <wan-protocol>pppoe</wan-protocol>
       <pppoe-user>dan</pppoe-user>
       <pppoe-password>dan</pppoe-password>
       <mgmt-prof>
        <type>OntRgMgmtProf</type>
        <id>
         <ontrgmgmtprof>1</ontrgmgmtprof>
        </id>
       </mgmt-prof>
       <tr69-eth-svc>
        <type>EthSvc</type>
        <id>
         <ont>1</ont>
         <ontslot>8</ontslot>
         <ontethany>1</ontethany>
         <ethsvc>1</ethsvc>
        </id>
       </tr69-eth-svc>
       <tr69-out-tag>none</tr69-out-tag>
```

```
       <tr69-in-tag>none</tr69-in-tag>
       <disable-on-batt>true</disable-on-batt>
       <pbit-map>
        <type>DscpMap</type>
        <id>
         <dscpmap>1</dscpmap>
        </id>
       </pbit-map>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="137" nodename="NTWK-412">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for provisioning an RG interface with the Mgmt mode configured as native and the WAN protocol configured as PPPoE.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the Remote Gateway (FB) port is the provisioning object type using the following case-sensitive expression - **OntRg**
<id> <ont> <ontslot> <ontrg>	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—The number **8** specifies an RG port <ontrg>—RG port number
<admin>	see descr		Configure the admin state as **enabled** to enable all traffic on the port.
<subscr-id>	Char(63)		Subscriber ID
<descr>	Char(31)		User description
<mgmt-mode>	see descr		The management mode is configured with the case sensitive word **native**. This is the default value when a new ONT is created. Native management mode allows the service provisioning from the E7. The Static and PPPoE parameters are applicable only in this case. The WAN protocol (dhcp, static, pppoe) is applicable only in the native management mode. ---------- The other available management mode is **external**. External management mode allows the RF service provisioning by an external RF configuration file or ACS. **Note:** T-Series ONTs do not support Native Mode as an option for the Management Mode.

Element Tag	Data Type	Req'd ?	Description
<wan-protocol>	see descr		The WAN protocol is configured with the case sensitive word **dhcp.**
<pppoe-user> <pppoe-password>	Char (31)		PPPoE user name and password
<mgmt-prof> <type> <id> <ontrgmgmtprof>	see descr		<type>- identifies the RG Management profile with the case sensitive word **OntRgMgmtProf** <ontrgmgmtprof>- identifies the RG Management profile number
<tr69-eth-svc> <type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr		<type> identifies the TR-69 Eth Service with the case sensitive word **EthSvc**. <ont>—identifies the ONT by its E7 scope ID (1 to 64000000) <ontslot>—The number **8** identifies the port type as an RG port. <ontethany>- identifies the ONT port number <ethsvc> - identifies the data service (1 to 12; typically 1 to 8 for data service).
<tr69-out-tag> <tr69-in-tag>	see descr		Configure the outer and inner tag VLAN ID tags for the access channel (**none**, **1** to **4093).** The content of these tags depends on the management profile used by the RG interface
<disable-on-batt>	Bool		Identifies the port operational state when the ONT is operating on battery backup power: • **false**—No (default if no tag is provided) • **true—Yes**

Element Tag	Data Type	Req'd ?	Description
<pbit-map> <type> <id> <dscpmap>	see descr		<type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpMap** <dscpmap> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.

Provision an RG interface in external mode

This section describes how to provision an RG interface with the Mgmt mode configured as external.

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc message-id="33" nodename="NTWK-AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="5">
            <edit-config>
                <target>
                    <running/>
                </target>
                <config>
                    <top>
                        <object operation="merge">
                            <type>OntRg</type>
                            <id>
                                <ont>1</ont>
                                <ontslot>8</ontslot>
                                <ontrg>1</ontrg>
                            </id>
                            <admin>enabled</admin>
                            <subscr-id>Test</subscr-id>
                            <descr></descr>
                            <mgmt-mode>external</mgmt-mode>
                            <config-file-instance>17</config-file-instance>
                            <mgmt-prof>
                                <type>OntRgMgmtProf</type>
                                <id>
                                    <ontrgmgmtprof>1</ontrgmgmtprof>
                                </id>
                            </mgmt-prof>
                            <tr69-eth-svc>
```

```
                                <type>EthSvc</type>
                                <id>
                                    <ont>1</ont>
                                    <ontslot>8</ontslot>
                                    <ontethany>1</ontethany>
                                    <ethsvc>1</ethsvc>
                                </id>
                            </tr69-eth-svc>
                            <tr69-out-tag>none</tr69-out-tag>
                            <tr69-in-tag>none</tr69-in-tag>
                            <disable-on-batt>true</disable-on-batt>
                            <pbit-map>
                                <type>DscpMap</type>
                                <id>
                                    <dscpmap>1</dscpmap>
                                </id>
                            </pbit-map>
                        </object>
                    </top>
                </config>
            </edit-config>
        </rpc>
    </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc-reply message-id="33" nodename="NTWK-AeCMSNetwork">
            <ok/>
        </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags for provisioning an RG interface with the Mgmt mode configured as external and the config file instance set.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the Remote Gateway (FB) port is the provisioning object type using the following case-sensitive expression - **OntRg**

Element Tag	Data Type	Req'd ?	Description
<id> <ont> <ontslot> <ontrg>	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—The number **8** specifies an RG port <ontrg>—RG port number
<admin>	see descr		Configure the admin state as **enabled** to enable all traffic on the port.
<subscr-id>	Char(63)		Subscriber ID
<descr>	Char(31)		User description
<mgmt-mode>	see descr		The management mode is configured with the case sensitive word **native**. This is the default value when a new ONT is created. Native management mode allows the service provisioning from the E7. The Static and PPPoE parameters are applicable only in this case. The WAN protocol (dhcp, static, pppoe) is applicable only in the native management mode. --- The other available management mode is **external**. External management mode allows the RF service provisioning by an external RF configuration file or ACS. **Note:** T-Series ONTs do not support Native Mode as an option for the Management Mode.
<config-file-instance>			
<mgmt-prof> <type> <id> <ontrgmgmtprof>	see descr		<type>- identifies the RG Management profile with the case sensitive word **OntRgMgmtProf** <ontrgmgmtprof>- identifies the RG Management profile number

Provision an FB interface

Sample XML request for provisioning an FB interface

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
```

```
<soapenv:Body>
 <rpc message-id="137" nodename="NTWK-412" timeout="35000" username="rootgod"
  sessionid="62">
  <edit-config>
   <target>
    <running/>
   </target>
   <config>
    <top>
     <object operation="merge">
      <type>OntFb</type>
      <id>
       <ont>1</ont>
       <ontslot>9</ontslot>
       <ontfb>1</ontfb>
      </id>
      <admin>enabled</admin>
      <subscr-id>707-766-3500</subscr-id>
      <descr>123 Main St</descr>
      <sec>
       <type>EthSecProf</type>
       <id>
        <ethsecprof>1</ethsecprof>
       </id>
      </sec>
      <pbit-map>
       <type>DscpMap</type>
       <id>
        <dscpmap>1</dscpmap>
       </id>
      </pbit-map>
     </object>
    </top>
   </config>
  </edit-config>
 </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

Sample XML response for provisioning an FB interface

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="137" nodename="NTWK-412">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>N
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for provisioning a GE port as an RG or HB interface.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntEthFB**—Full Bridge (FB) port
<id> <ont> <ontslot> <ontfb>	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—port type: • **9**—FB port <ontfb>—FB port number
<admin>	see descr		Configure the admin state as **enabled** to enable all traffic on the port
<subscr-id>	Char(63		Subscriber ID
<descr>	Char(31)		User description
<sec> <type> <id> <ethsecprof>	see descr		<type> identifies the provisioning object as an ONT security profile using the following case-sensitive expression: **EthSecProf** <ethsecprof> identifies the global Ethernet Security profile ID (**1** to **16**).
<pbit-map> <type> <id> <dscpmap>	see descr		<type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpMap** <dscpmap> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.

Update an E7 ONT Ethernet port

Sample XML request for configuring HSI service

The following is an example of an update (merge) request for ONT Ethernet port 1.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="222" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="merge">
              <type>OntEthGe</type>
              <id>
                <ont>101222</<ont>
                <ontslot>3</ontslot>
                <ontethge>1</onethge>
              </id>
              <admin>enabled</admin>
              <subscr-id>707-766-3500</subscr-id>
              <descr>123 Main St</descr>
              <gos>
                <type>OntEthPortGos</type>
                <id>
                  <ontethportgos>1</ontethportgos>
                </id>
              </gos>
              <duplex>full</duplex>
              <sec>
                <type>EthSecProf</type>
                <id>
                  <ethsecprof>1</ethsecprof>
                </id>
              </sec>
              <disable-on-batt>false</disable-on-batt>
              <link-oam-events>false</link-oam-events>
              <accept-link-oam-loopbacks>false</accept-link-oam-loopbacks>
              <speed>auto</speed>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **merge** request for adding or updating the subscriber ID and user description on an E7 GPON port. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
`<type>`	See descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: • **OntEthGe**—GE port

`<id>`
`<ont>`
`<ontslot>`
`<ontethge>`
	see descr	Yes	These element tags identify the port:
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—port type:
			• **3**—Gigabit Ethernet port
			`<ontethge>`—port number

`<admin>`	Char		Use the following: • **enabled**—enable all traffic on the port
`<subscr-id>`	Char(63)		Subscriber ID
`<descr>`	Char(47)		User description

`<gos>`
`<type>`
`<id>`
`<ontethportgos>`
	Char		`<type>` identifies the provisioning object as an ONT Ethernet port grade-of-service (GOS) profile using the following case-sensitive expression:
			• **OntEthPortGos**
			`<ontethportgos>` identifies the global GOS profile ID (**1** to **10**).

`<duplex>`	Bool		Identifies the duplex mode for the Ethernet port: • **full** (default if no tag is provided) • **half**

`<sec>`
`<type>`
`<id>`
`<ethsecprof>`
	Char		`<type>` identifies the provisioning object as an ONT security profile using the following case-sensitive expression: **EthSecProf**
			`<ethsecprof>` identifies the global Ethernet Security profile ID (**1** to **16**).

Element Tag	Data Type	Req'd ?	Description
<disable-on-batt>	Bool		Identifies the port operational state when the ONT is operating on battery backup power: • **false**—No (default if no tag is provided) • **true**—Yes
<link-oam-events>	Bool		Identifies whether to enable OAM event monitoring: • **false**—Disabled (default if no tag is provided) • **true**—Enabled
<accept-link-oam-loopbacks>	Bool		Identifies whether the port accepts or rejects 802.3ah frames sent by the host: • **false**—No/reject (default if no tag is provided) • **true—**Yes/accept
<speed>	Char		Identifies the data rate of the Ethernet port: • **auto**—auto negotiate the rate (default if no tag is provided) • **1000**—1000Mbps

Note: If a port parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply.

E7 GPON ONT Data Service Activation

The XML request defines the objects required to create, update, and delete data services on an E7 GPON ONT Ethernet port.

This section contains the following topics:

For read (get-config) requests, see "XML Read Requests for GPON Service Activation" on page 125.

Create data service on E7 GPON ONTs

Sample XML create request

Note: In the following example, XML tags for outer and inner VLANs are not supplied.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="217" nodename="NTWK-WestE7" username="JDoe"
         sessionid="37">
      <edit-config><target><running/></target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>EthSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>5</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>3</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
                  <bwprof>1</bwprof>
                </id>
              </bw-prof>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating data service on an E7 GPON ONT. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port. <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> - identifies the data service (1 to 12; typically 1 to 8 for data service).
<admin>	Char		Operational status of the subscriber port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<descr>	Char(31)		Description of service.
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (**1** to **255**).

Element Tag	Data Type	Req'd?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a global Ethernet bandwidth profile (**1** to **300**).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>EthSvc</type>
             <id>
              <ont name="">1</ont>
              <ontslot>5</ontslot>
              <ontethany>1</ontethany>
              <ethsvc name="Data1">1</ethsvc>
            </id>
            <admin>enabled</admin>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction name="Test_data_svc" localId="24">3</svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name="Test_bw_profile" localId="12">1</bwprof>
              </id>
            </bw-prof>
            <descr/>
            <out-tag>none</out-tag>
            <in-tag>none</in-tag>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 168). **Note:** The local IDs of the Service Tag Action and Ethernet Bandwidth profiles that are mapped to the global profile are returned in the XML reply. See the above reply for examples.

Update (merge) data service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) data service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>EthSvc</type>
                . . .
```

Note: The element tags that can be included in a **merge** request for data service on an E7 GPON ONT are the same as the input element tags (see "XML input element tags and values" on page 168). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 168.

Delete data service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="183" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>EthSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>5</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** data service request for an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port. <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the data service (**1** to **12**; typically **1** to **8** for data service).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="183" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** data service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

E7 GPON ONT Video Service Activation

The XML request defines the objects required to create, update, and delete video services on an E7 GPON ONT Ethernet port.

This section contains the following topics:

For read (get-config) requests, see "XML Read Requests for GPON Service Activation" on page 125.

Multicast VLAN Registration (MVR) video service

To activate MVR video service, an MVR profile must already be created and configured on the E7. The MVR profile is referenced by a multicast profile, which is a required element tag in setting up E7 video service.

Create video service on E7 GPON ONTs

Sample XML create request on ONT FB Port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename=”NTWK-WestE7” username=”JDoe”
   sessionid=”83”>
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>EthSvc</type>
       <id>
        <ont>1</ont>
        <ontslot>9</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr></descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>16</svctagaction>
        </id>
        </tag-action>
        <bw-prof>
```

```
          <type>BwProf</type>
          <id>
           <bwprof>24</bwprof>
          </id>
         </bw-prof>
         <out-tag>101</out-tag>
         <in-tag>none</in-tag>
         <mcast-prof>
          <type>McastProf</type>
          <id>
           <mcastprof>5</mcastprof>
          </id>
         </mcast-prof>
         <pon-cos>derived</pon-cos>
        </object>
       </top>
      </config>
     </edit-config>
   </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating video service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port. • **8**—RG port. • **9**—FB port. <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service).
<descr>	Char(31)		Description of service.
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (1 to 255).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies an ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).
<pon-cos>	see descr		Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="226" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>EthSvc</type>
             <id>
              <ont name="">1</ont>
              <ontslot>5</ontslot>
              <ontethany>1</ontethany>
              <ethsvc name="Data1">1</ethsvc>
            </id>
            <admin>enabled</admin>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction name="Test_data_svc" localId="24">3
                     </svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name="Test_bw_prof" localId="12">1</bwprof>
              </id>
            </bwprof>
            <descr/>
            <out-tag>none</out-tag>
            <in-tag>none</in-tag>
            <mcast-prof>1</mcast-prof>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 175). **Note:** The local IDs of the Service Tag Action, Ethernet bandwidth, and Multicast profiles that are mapped to the global profiles are returned in the XML reply. See the above reply for examples.

Update (merge) video service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) video service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="158" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>EthSvc</type>
                . . .
```

Note: The element tags that can be included in a **merge** request for video service on an E7 GPON ONT are the same as the input element tags (see "XML input element tags and values" on page 175). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 175).

Delete video service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="198" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>EthSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>5</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>9</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** video service request on an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port. • **8**—RG port. • **9**—FB port. <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="198" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** video service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Enable and Disable RF Video and RF Return on E7 GPON ONTs

Sample XML merge request for enabling RF video

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="236" nodename="NTWK-WestE7" username="JDoe"
        sessionid="85">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntVideoRf</type>
              <id>
                <ont>1</ont>
                <ontslot>1</ontslot>
                <ontvideorf>1</ontvideorf>
              </id>
              <admin>enabled</admin>
              <subscr-id>707-766-3500</subscr-id>
              <descr>123 Main St</descr>
              <disable-on-batt>false</disable-on-batt>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML merge request for enabling RF return

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="237" nodename="NTWK-WestE7" username="JDoe"
        sessionid="85">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntRfAvo</type>
              <id>
                <ont>1</ont>
                <ontrfavo>1</ontrfavo>
              </id>
              <admin>enabled</admin>
```

```
            <rf-return-state>enabled</rf-return-state>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for enabling RF video on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontvideorf>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: • **OntVideoRf** <ont> identifies the ONT by its E7 scope ID (1 to 64000000). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the port type using one of the following: • **1**—RF-video port • **2**—Hot RF-video port <ontvideorf> identifies the RF-video or hot RF video port (1 to 8).
<admin>	Char	Yes	Administrative state of the port: • **disabled** • **enabled** • **enabled-no-alarms**
<subscr-id>	Char(27)		Subscriber ID information
<descr>	Char(27)		Description of the port or service
<disable-on-batt>	Bool		Port operational state when the ONT is operating on battery backup power: • **true—**Disable • **false—**Do not disable (default)

The following table lists the XML element tags in a **merge** request for enabling RF return on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontrfavo>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: • **OntRfAvo** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontrfavo> identifies the AVO port (**1**).
<rf-return-state>	Char	Yes	Identifies the operational status of the RF return path: • **disabled** • **enabled**
<admin>	Char		Identifies the operational status of the port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Sample XML reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id=”208” nodename=”NTWK-WestE7”>
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

E7 GPON ONT Voice Service Activation

The XML request defines the objects required to create, update, and delete SIP (SIP Remote, SIP Gateway), TDM Gateway, and H.248 VoIP services on E7 GPON ONT voice ports.

Before voice service can be activated, you must update the IP host for each type of voice service (SIP, TDM Gateway, H.248 Gateway, or MGCP Gateway) for each E7 GPON ONT.

This section contains the following topics:

For read (get-config) requests, see "XML Read Requests for GPON Service Activation" on page 125.

Update (merge) an IP host for VoIP service

Sample XML merge requests

The following example is for an IP host for SIP service using DHCP host protocol.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="333" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>OntIpHost</type>
              <id>
                <ont>4</ont>
                <ontiphost>1</ontiphost>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>18</svctagaction>
                </id>
              </tag-action>
              <host-proto>dhcp</host-proto>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example is for an IP host for TDM Gateway service with static host protocol.

Note: When creating a TDM Gateway service, the OntIpHost field is required.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="334" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>OntIpHost</type>
              <id>
                <ont>4</ont>
                <ontiphost>2</ontiphost>
              </id>
```

```
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction>5</svctagaction>
              </id>
            </tag-action>
            <host-proto>static</host-proto>
            <static-ip>192.11.14.101</static-ip>
            <static-mask>255.255.255.0</static-mask>
            <static-gw>192.11.14.102</static-gw>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **merge** request for configuring an IP host for E7 GPON ONT voice service.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontiphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntIpHost** <ont> identifies the ONT by its E7 scope ID (1 to 64000000). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontiphost> identifies the VoIP service type using one of the following: • **1**—SIP service • **2**—TDM Gateway service • **4**—H.248 Gateway service • **5**—MGCP Gateway service **Note:** When creating a TDM Gateway service, the OntIpHost field is required.

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action.
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<host-proto>	see descr	Yes	Use one of the following: • **dhcp**—DHCP host protocol • **static**—Static host protocol • **unknown**
Additional element tag for H.248 and MGCP with DHCP host protocol: <hostname>	Char	see descr	Fully qualified domain name of the DHCP host server, required for H.248 and MGCP service when <hostprotocol> is **dhcp**.
Additional element tags for static host protocol: <static-ip> <static-mask> <static-gw>	see descr	Yes	Use the following for these three additional element tags: • static IP address assigned to the ONT, in dotted quad format. • static IP mask assigned to the ONT, in dotted quad format. • static IP gateway for the ONT to use in routing traffic, in dotted quad format (must belong to the same subnet as the subtending IP address) **Note:** When the host protocol is DHCP, these element tags are not used (though the values are saved in the service record).

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 186). **Note:** The local ID of the service tag action profile that is mapped to the global profile is returned in the XML reply.

SIP VoIP Service on E7 GPON ONTs

There are two types of SIP VoIP service:

- **SIP Gateway:** Requires provisioning of the SIP gateway and the SIP dial plan on the E7.
  Note: The
- **SIP Remote:** Requires provisioning of a SIP configuration file stored on a TFTP server with a remote profile pointing to the file and location.

Create Gateway SIP VoIP service on E7 GPON ONTs

Before Gateway SIP voice service can be activated for the first time, complete the following pre-requisites:

- Update the IP host on the E7 GPON ONT. See "Update (merge) an IP host for VoIP service" on page 185.
- Provision the applicable E7 profiles.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
   username="rootgod" sessionid="16">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>SipSvc</type>
       <id>
        <ont>1</ont>
        <ontslot>6</ontslot>
        <ontpots>1</ontpots>
        <sipsvc>1</sipsvc>
       </id>
       <user>test</user>
       <passwd>test</passwd>
       <uri>test</uri>
       <admin>enabled</admin>
       <sip-prof>
```

```
        <type>SipGwProf</type>
        <id>
         <sipgwprof>2</sipgwprof>
        </id>
       </sip-prof>
       <call-waiting>true</call-waiting>
       <caller-id-enabled>true</caller-id-enabled>
       <three-way-calling>true</three-way-calling>
       <t38-fax-relay>false</t38-fax-relay>
       <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
       <dial-plan>
        <type>DialPlan</type>
        <id>
         <dialplan>1</dialplan>
        </id>
       </dial-plan>
       <enable-msg-waiting-ind>true</enable-msg-waiting-ind>
       <direct-connect></direct-connect>
       <direct-conn-timer-sec>0</direct-conn-timer-sec>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM Gateway VoIP service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **SipSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). **Note:** The number of voice ports varies depending on the ONT model. <sipsvc> identifies the voice service (**1**).

Element Tag	Data Type	Req'd ?	Description
<sip-prof> <type> <id> <sipgwprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **SipGwProf** <sipgwprof> identifies the ID of a pre-defined global SIP Gateway profile (**1** to **512**). The SIP Gateway profile resides on the E7 and is gotten by the Voice IP host over OMCI.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<user>	Char	Yes	User name.
<passwd>	Char	Yes	Password for specified user.
<uri>	Char	Yes	Universal resource identifier for SIP service.
<call-waiting> <caller-id-enabled> <three-way-calling> <t38-fax-relay>	Bool.		These parameters apply to E7 platforms operating with E7 R2.1.40 firmware or higher. Valid values are: • **true—**Enabled • **false**—Disabled (default) <call-waiting> Call Waiting feature <caller-id-enabled> Caller ID feature <three-way-calling> Three-Way Calling feature <t38-fax-relay> T38 Fax Relay feature
<sip-rmt-cfg-override>	Bool.		Override flag for SIP remote service configuration. Applies to GPON ONTs operating with R2.1.40 or higher firmware: • **true—**use the SIP configuration file • **false**—use the local configuration (default)

Sample XML create reply

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
   <ok/>
   <data>
    <top>
     <object>
```

```
        <type>SipSvc</type>
        <id>
         <ont>1</ont>
         <ontslot>6</ontslot>
         <ontpots>1</ontpots>
         <sipsvc>1</sipsvc>
        </id>
        <admin>enabled</admin>
        <sip-prof>
         <type>SipGwProf</type>
         <id>
          <sipgwprof name="@dd_sip_gw">2</sipgwprof>
         </id>
        </sip-prof>
        <user>test</user>
        <min>0</min>
        <warn>0</warn>
        <passwd>test</passwd>
        <uri>test</uri>
        <call-waiting>true</call-waiting>
        <caller-id-enabled>true</caller-id-enabled>
        <derived-states>parent-disabled</derived-states>
        <svc-status/>
        <call-state/>
        <dhcp-attempts/>
        <dhcp-acks/>
        <dhcp-nacks/>
        <reg-attempts/>
        <three-way-calling>true</three-way-calling>
        <t38-fax-relay>false</t38-fax-relay>
        <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
        <dial-plan>
         <type>DialPlan</type>
         <id>
          <dialplan name="system-default" localId="1"/>
         </id>
        </dial-plan>
        <direct-connect/>
        <direct-conn-timer-sec>0</direct-conn-timer-sec>
        <enable-msg-waiting-ind>true</enable-msg-waiting-ind>
        <op-stat>sys-disable</op-stat>
        <crit>0</crit>
        <maj>0</maj>
        <info>0</info>
        <hook-state/>
        <config-status/>
        <reg-challenges/>
       </object>
      </top>
     </data>
    </rpc-reply>
   </soapenv:Body>
  </soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 189). **Note:** The local ID of the SIP profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Create Remote SIP VoIP service on E7 GPON ONTs

Before Gateway SIP voice service can be activated for the first time, complete the following pre-requisites:

- Update the IP host on the E7 GPON ONT. See "Update (merge) an IP host for VoIP service" on page 185.
- Provision the SIP remote configuration file and place it on the tftp server.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="240" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>SipSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <sipsvc>1</sipsvc>
              </id>
              <sip-prof>
                <type>SipProf</type>
                <id>
                  <sipprof>1</sipprof>
                </id>
              </sip-prof>
              <admin>enabled</admin>
              <user>West_sip_svc</user>
              <passwd>test</passwd>
              <uri>uri</uri>
              <call-waiting>false</call-waiting>
              <caller-id-enabled>true</caller-id-enabled>
```

```
            <three-way-calling>true</three-way-calling>
            <t38-fax-relay>false</t38-fax-relay>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM Gateway VoIP service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **SipSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). **Note:** The number of voice ports varies depending on the ONT model. <sipsvc> identifies the voice service (**1**).
<sip-prof> <type> <id> <sipprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **SipProf** <sipprof> identifies the ID of a pre-defined global SIP profile (**1** to **512**). The SIP remote profile is the external ASCII configuration file downloaded from a TFTP site.

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<user>	Char	Yes	User name.
<passwd>	Char	Yes	Password for specified user.
<uri>	Char	Yes	Universal resource identifier for SIP service.
<call-waiting> <caller-id-enabled> <three-way-calling> <t38-fax-relay>	Bool.		These parameters apply to E7 platforms operating with E7 R2.1.40 firmware or higher. Valid values are: • **true—**Enabled • **false**—Disabled (default) <call-waiting> Call Waiting feature <caller-id-enabled> Caller ID feature <three-way-calling> Three-Way Calling feature <t38-fax-relay> T38 Fax Relay feature
<sip-rmt-cfg-override>	Bool.		Override flag for SIP remote service configuration. Applies to GPON ONTs operating with R2.1.40 or higher firmware: • **true—**use the SIP configuration file • **false**—use the local configuration (default)

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id=”240” nodename=”NTWK-WestE7”
      <ok/>
      <data>
        <top>
          <object>
            <type>SipSvc</type>
            <id>
              <ont name=””>1</ont>
              <ontslot>6</ontslot>
              <ontpots>2</ontpots>
              <sipsvc>1</sipsvc>
            </id>
            <admin>enabled</admin>
```

```
            <sip-prof>
              <type>SipProf</type>
              <id>
                <sipprof name="Test_sip_gw_prof" localId="4">1</sipprof>
              </id>
            </sip-prof>
            <user>Test_sip_svc</user>
            <passwd>test</passwd>
            <uri>uri</uri>
            <call-waiting>false</call-waiting>
            <caller-id-enabled>true</caller-id-enabled>
            <three-way-calling>true</three-way-calling>
            <t38-fax-relay>false</t38-fax-relay>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 189). **Note:** The local ID of the SIP profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) SIP voice service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (**merge**) SIP voice service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="230" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
```

```
            <object operation="merge" get-config="true">
              <type>SipSvc</type>
                . . .
```

Delete SIP VoIP service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
     <rpc message-id="96" nodename="NTWK-AeCMSNetwork" timeout="35000"
      username="rootgod" sessionid="22">
      <edit-config>
        <target>
         <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>SipSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <sipsvc>1</sipsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
    <soapenv:Body>
        <rpc-reply message-id="96" nodename="NTWK-AeCMSNetwork">
            <ok/>
        </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123. The following table lists the XML element tags required in a **delete** SIP VoIP service request on an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **SipSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). <sipsvc> identifies the voice service (**1**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="806" nodename="NTWK-WestE7">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** SIP VoIP service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

TDM Gateway VoIP service on E7 GPON ONTs

Create TDM Gateway VoIP service on E7 GPON ONTs

Before TDM Gateway voice service can be activated for the first time, you must update the IP host on the E7 GPON ONT. See "Update (merge) an IP host for VoIP service" on page 185.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1023" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>TdmGwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>3</ontpots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
              <tdmgw-prof>
                <type>TdmGwProf</type>
                <id>
                  <tdmgwprof>1</tdmgwprof>
                </id>
              </tdmgw-prof>
              <tdmgw-svc-group>
                <type>TDMGWServiceGroup</type>
                <id>
                  <tdmgwservicegroup>8</tdmgwservicegroup>
                </id>
              </tdmgw-svc-group>
              <admin>enabled</admin>
              <crv>N1-1-IG1-224</crv>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM Gateway VoIP service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **TdmGwSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). **Note:** The number of ONT voice ports varies based on the ONT model. <tdmgwsvc> identifies the voice service (**1**).
<tdmgw-prof> <type> <id> <tdmgwprof>	see descr	Yes	<type> identifies the profile type as TDM Gateway using the following case-sensitive expression: **TdmGwProf** <tdmgwprof> identifies the ID of a pre-defined global TDM Gateway profile (**1** to **32**).
<tdmgw-svc-group> <type> <id> <tdmgwservicegroup>	see descr	Yes	<type> identifies the object type using the following case-sensitive expression: **TDMGWServiceGroup** <tdmgwservicegroup> identifies the ID of a pre-defined CMS TDM Gateway Service group (**1** to **1000**).

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<crv>	Char	Yes	Call Reference Value for the subscriber line in GR-303 or GR-8 switch interface group. Example: **N1-1-IG1-224** Note the following: • The CRV must be provisioned on the C7 gateway interface group. • The format is case-sensitive, and must be upper case.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id=”1023” nodename=”NTWK-WestE7”
      <ok/>
      <data>
        <top>
          <object>
            <type>TdmGwSvc</type>
            <id>
              <ont name=””>1</ont>
              <ontslot>6</ontslot>
              <ontpots>3</ontpots>
              <tdmgwsvc>1</tdmgwsvc>
            </id>
            <admin>enabled</admin>
            <crv>N1-1-IG1-224</crv>
            <tdmgw-prof>
              <type=TdmGwProf</type>
              <id>
               <tdmgwprof name=”Test_tdm_gw_profile” localId=”5”>1<tdmgwprof>
              </id>
            </tdmgw-prof>
```

```
            <tdmgw-svc-group>
              <type>TDMGWServiceGroup</type>
              <id>
                <tdmgwservicegroup name= ”tdm_svc_grp5”>8</tdmgwservicegroup>
              </id>
            </tdmgw-svc-group>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see “Common XML Element Tags” on page 123.

If the <object> tag in the request includes the attribute, get-config=“true”, element tags for provisioning parameters are returned in the reply (for descriptions, see “XML input element tags and values” on page 199). **Note:** The local ID of the TDM Gateway profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) TDM Gateway voice service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) TDM Gateway voice service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id=”231” nodename=”NTWK-WestE7” username=”JDoe”
        sessionid=”65”>
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config=”true”>
              <type>TdmGwSvc</type>
                . . .
```

Delete TDM Gateway VoIP service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="810" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>TdmGwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** TDM Gateway VoIP service request on an E7 GPON ONT.

Element Tag	Data Type	Req'd?	Description
<type> <id> <ont> <ontslot> <ontpots> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **TdmGwSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). <tdmgwsvc> identifies the service type (**1**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="810" nodename="NTWK-WestE7">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** TDM Gateway VoIP service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123

H.248 VoIP service on E7 GPON ONTs

Create H.248 VoIP service on E7 GPON ONTs

Before H.248 voice service can be activated for the first time, you must update the IP host on the E7 GPON ONT. See "Update (merge) an IP host for VoIP service" on page 185.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1130" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>H248GwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <h248gwsvc>1</h248gwsvc>
              </id>
              <h248gw-prof>
                <type>H248GwProf</type>
                <id>
                  <h248gwprof>1</h248gwprof>
                </id>
              </h248gw-prof>
              <termination-id>1</termination-id>
              <admin>enabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating H.248 VoIP service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontds1> <h248gwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **H248GwSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). N**ote:** The number of voice ports varies depending on the ONT model. <h248gwsvc> identifies the service type (**1**).
<h248gw-prof> <type> <id> <h248gwprof>	see descr	Yes	<type> identifies the profile as an H.248 type using the following case-sensitive expression: **H248GwProf** <h248gwprof> identifies the ID of a pre-defined global H.248 profile (1 to 20).
<termination-id>	Char	Yes	Identifies the H.248 termination ID.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1130" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>H248GwSvc</type>
            <id>
```

```
            <ont name="">1</ont>
            <ontslot>6</ontslot>
            <ontpots>4</ontpots>
            <h248gwsvc>1</h248gwsvc>
          </id>
          <admin>enabled</admin>
          <h248gw-prof>
            <type=H248GwProf</type>
            <id>
              <h248gwprof name="Test_h.248_prof" localId="4">1<h248gwprof>
            </id>
          </h248gw-prof>
          <termination-id>1</termination-id>
        </object>
      </top>
    </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 204). **Note:** The local ID of the H.248 Gateway profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) H.248 voice service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (**merge**) H.248 VoIP service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="232" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>H248GwSvc</type>
                . . .
```

Delete H.248 VoIP service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="811" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>H248GwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <h248gwsvc>1</h248gwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** H.248 VoIP service request on an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <h248gwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **H248GwSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**).The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). <h248gwsvc> identifies the service type (**1** to **20**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="811" nodename="NTWK-WestE7">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** H.248 VoIP service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

MGCP VoIP service on E7 GPON ONTs

Create MGCP VoIP service on E7 GPON ONTs

Before MGCP voice service can be activated for the first time, you must update the IP host on the E7 GPON ONT. See "Update (merge) an IP host for VoIP service" on page 185.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1130" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>MgcpGwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <mgccpgwsvc>1</mgcpgwsvc>
              </id>
              <mcgpgw-prof>
                <type>MgcpGwProf</type>
                <id>
                  <mgcpgwprof>1</mgcpgwprof>
                </id>
              </mgcpgw-prof>
              <termination-id>1</termination-id>
              <admin>enabled</admin>
              <gr-303>false</gr-303>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating MGCP VoIP service on an E7 GPON ONT.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontds1> <mgcpgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **MgcpGwSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). N**ote:** The number of voice ports varies depending on the ONT model. <mgcpgwsvc> identifies the service type (**1**).
<mgcpgw-prof> <type> <id> <mgcpgwprof>	see descr	Yes	<type> identifies the profile as an MGCP type using the following case-sensitive expression: **MgcpGwProf** <mgcpgwprof> identifies the ID of a pre-defined global H.248 profile (**1** to **20**).
<termination-id>	Char	Yes	Identifies the MGCP termination ID.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<gr-303>	Bool		(Requires E7 platform running R2.1.40 or higher.) Enables or disables RFC 2833 voice control protocol compatible with the B6-640 Trunking Gateway (TGW) in GR-303 mode. Valid values: • **true—**Enabled • **false**—Disabled (default)

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1130" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>MgcpGwSvc</type>
            <id>
              <ont name="">1</ont>
              <ontslot>6</ontslot>
              <ontpots>4</ontpots>
              <mgcpgwsvc>1</mgcpgwsvc>
            </id>
            <admin>enabled</admin>
            <mgcpgw-prof>
              <type=MgcpGwProf</type>
              <id>
                <mgcpgwprof name="Test_MGCP_prof" localId="4">1<mgcpgwprof>
              </id>
            </mgcpgw-prof>
            <termination-id>1</termination-id>
            <gr-303>false</gr-303>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 209). **Note:** The local ID of the MGCP Gateway profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) MGCP voice service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (**merge**) MGCP VoIP service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="232" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>MgcpGwSvc</type>
                . . .
```

Delete MGCP VoIP service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="811" nodename="NTWK-WestE7" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>MgcpGwSvc</type>
              <id>
                <ont>1</ont>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <mgcpgwsvc>1</mgcpgwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** MGCP VoIP service request on an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontpots> <mgcpgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **MgcpGwSvc** <ont> identifies the ONT by its E7 scope ID (1 to 64000000).The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type as voice (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). <mgcpgwsvc> identifies the service type (**1** to **20**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="811" nodename="NTWK-WestE7">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** MGCP VoIP service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123

E7 GPON ONT PWE3 Service Activation

The XML request defines the objects required to create, update, and delete PWE3 services on an E7 GPON ONT T1 port.

Before voice service can be activated, you must update the IP host for PWE3 service for each E7 GPON ONT.

This section contains the following topics:

For read (get-config) requests, see "XML Read Requests for GPON Service Activation" on page 125.

Update (merge) an IP host for PWE3 DS1 service

Sample XML merge request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="334" nodename="NTWK-WestE7" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>OntIpHost</type>
              <id>
                <ont>4</ont>
                <ontiphost>4</ontiphost>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>5</svctagaction>
                </id>
              </tag-action>
              <out-tag>none</out-tag>
              <in-tag>none</in-tag>
              <host-proto>static</host-proto>
              <static-ip>192.11.14.101</static-ip>
              <static-mask>255.255.255.0</static-mask>
              <static-gw>192.11.14.102</static-gw>
            </object>
          </top>
        </config>
```

```
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for configuring a static host for E7 GPON ONT voice service.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontiphost>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntIpHost** <ont> identifies the ONT by its E7 scope ID (1 to 64000000). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontiphost> identifies the service type using the following: • **3**—PWE3 service
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies an ID of a pre-defined global service tag action (1 to 255).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tags in the XML request. If not supplied, the XML reply returns the value **none**.
<hostprotocol>	see descr	Yes	Use the following: • **static**—Static host protocol

Element Tag	Data Type	Req'd ?	Description
<static-ip> <static-mask> <static-gw>	see descr	Yes	Use the following for these three additional element tags: • static IP address assigned to the ONT, in dotted quad format. • static IP mask assigned to the ONT, in dotted quad format. • static IP gateway for the ONT to use in routing traffic, in dotted quad format (must belong to the same subnet as the subtending IP address) **Note:** When the host protocol is DHCP, these element tags are not used (though the values are preserved in the service record).

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 215). **Note:** The local ID of the Service Tag Action profile that is mapped to the global profile is returned in the XML reply.

Create PWE3 service on E7 GPON ONTs

Sample XML create request

The following example shows an XML request using the UDP-IP transport mode.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1414" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>Pwe3Svc</type>
              <id>
                <ont>6</ont>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
              <ds1-pwe3-prof>
                <type>Ds1Pwe3Prof</type>
                <id>
                  <ds1pwe3prof>4</ds1pwe3prof>
                </id>
              </ds1-pwe3-prof>
              <admin>enabled</admin>
              <transport>udp-ip</transport>
              <udp-port>4566</udp-port>
              <remote-udp>4566</remote-udp>
              <remote-ip>191.23.43.33</remote-ip>
              <remote-mac>00:00:00:00:00:00</remote-mac>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML request using the MEF transport mode.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1414" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
```

```
            <object operation="create" get-config="true">
              <type>Pwe3Svc</type>
              <id>
                <ont>6</ont>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
              <ds1-pwe3-prof>
                <type>Ds1Pwe3Prof</type>
                <id>
                  <ds1pwe3prof>4</ds1pwe3prof>
                </id>
              </ds1-pwe3-prof>
              <admin>enabled</admin>
              <transport>mef</transport>
              <remote-ip>191.23.43.33</remote-ip>
              <ecid>3300</ecid>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating PWE3 service on an E7 GPON ONT. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontds1> <pwe3svc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Pwe3Svc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**).The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type (**7**). <ontds1> identifies the ONT T1/E1 port number (**1** to **8**). N**ote:** The number of T1/E1 ports varies depending on the ONT model. <pwe3svc> identifies the service type (**1**).

Element Tag	Data Type	Req'd ?	Description
<ds1-pwe3-prof> <type> <id> <ds1pwe3prof>	see descr	Yes	<type> identifies the profile type as DS1 PWE3 using the following case-sensitive expression: **Ds1Pwe3Prof** <ds1pwe3prof> identifies the ID of a pre-defined global T1/E1 PWE3 profile (**1** to **8**).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<transport>	Char		Identifies the transport mode using one of the following: • **udp-ip** (default, if element tag is not supplied) • **mef**
<udp-port>	Int	in UPD-IP mode	(Required in UDP-IP transport mode) Local UDP port number (**1024** to **65534**).
<remote-udp>	Int		(Optional in UDP-IP transport mode) Remote UDP port number (**1024** to **65534**).
<remote-IP>	IP address format	see note	IP address of the remote end of the pseudowire, in dotted quad format. Alternately, supply **none** to reset the value to 0.0.0.0. **Note:** In UDP-IP transport mode, this element tag is required. In MEF transport mode, either this element tag or the remote-mac element tag is required.
<remote-mac>	MAC format	see note	(Applicable for MEF transport mode) The MAC address of the remote end of the pseudowire. The data format is six hexadecimal digits in the range 0-FF, optionally separated by colons. **Note:** In MEF transport mode, either this element tag or the remote-ip element tag must be supplied.
<ecid>	Int		(Optional in MEF transport mode) Emulated Circuit ID (**0** to **1048575**).
<remote-ecid>	Int		(Optional in MEF transport mode) Remote Emulated Circuit ID (0 to 1048575).

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs), when required for the service. **Note:** If an outer or inner VLAN is not required, use 4095, or do not include the element tags in the XML request.

Sample XML create reply

The following example shows an XML reply for UDP-IP transport mode.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1414" nodename="NTWK-WestE7"
      <ok/>
      <data>
        <top>
          <object>
            <type>Pwe3Svc</type>
             <id>
              <ont name="">1</ont>
              <ontslot>7</ontslot>
              <ontds1>1</ontds1>
              <pwe3svc>1</pwe3svc>
            </id>
            <admin>enabled</admin>
            <ds1-pwe3-prof>
              <type>Ds1Pwe3Prof</type>
              <id>
                <ds1pwe3prof name="Test_pwe3_profile" localId="6">1
                   </ds1pwe3prof>
              </id>
            </ds1-pwe3-prof>
            <transport>udp-ip</transport>
            <udp-port>4566</udp-port>
            <remote-udp>4566</remote-udp>
            <remote-ip>191.23.43.33</remote-ip>
            <remote-mac>00:00:00:00:00:00</remote-mac>
            <ecid>0</ecid>
            <remote-ecid>0</remote-ecid>
            <gos>
              <type>OntPwe3Gos</type>
              <id>
                <ontpwe3gos>1</ontpwe3gos>
              </id>
            </gos>
          </object>
        </top>
      </data>
    </rpc-reply>
```

```
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML reply for MEF transport mode.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1414" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create" get-config="true">
              <type>Pwe3Svc</type>
              <id>
                <ont>6</ont>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
              <ds1-pwe3-prof>
                <type>Ds1Pwe3Prof</type>
                <id>
                  <ds1pwe3prof name="Test_pwe3_profile" localId="6">4
                     </ds1pwe3prof>
                </id>
              </ds1-pwe3-prof>
              <admin>enabled</admin>
              <transport>mef</transport>
              <remote-ip>191.23.43.33</remote-ip>
              <ecid>3300</ecid>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 218). **Note:** The local ID of the DS1 PWE3 profile that is mapped to the global profile is returned in the XML reply. See the above reply for an example.

Update (merge) PWE3 service on E7 GPON ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) PWE3 service on an E7 GPON port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1415" nodename="NTWK-WestE7" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge" get-config="true">
              <type>Pwe3Svc</type>
                . . .
```

Note: The element tags in a **merge** request for PWE3 service on an E7 GPON ONT are the same as the input element tags (see "XML input element tags and values" on page 218). If a service parameter value is not changing, the corresponding XML element tag does not have to be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

If the <object> tag in the request includes the attribute, get-config="true", element tags for all provisioning parameters are returned in the reply (for descriptions, see "XML input element tags and values" on page 218).

Delete PWE3 service on E7 GPON ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1416" nodename="NTWK-WestE7" nodename="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>Pwe3Svc</type>
              <id>
                <ont>6</ont>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required in a **delete** PWE3 service request for an E7 GPON ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontds1> <pwe3svc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Pwe3Svc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the ONT port type (**7**). <ontds1> identifies the ONT T1/E1 port number (**1** to **8**). N**ote:** The number of T1/E1 ports varies depending on the ONT model. <pwe3svc> identifies the service type (**1**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1416" nodename="NTWK-WestE7">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** PWE3 service request for an E7 GPON ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

E7 GPON ONT Gateway Service Activation on an RG Port

The XML request defines the objects required to create, update, and delete a gateway service on an E7 GPON ONT RG port.

This section contains the following topics:

For read (get-config) requests, see "XML Read Requests for GPON Service Activation" on page 125.

Create a gateway service on an RG port on a E7 GPON ONT

Sample XML create gateway service request on ONT RG Port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename="NTWK-WestE7" username="JDoe"
   sessionid="83">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>EthSvc</type>
       <id>
        <ont>1</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr></descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>158</svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
        <id>
         <bwprof>10</bwprof>
        </id>
       </bw-prof>
       <out-tag>100</out-tag>
```

```
       <in-tag>none</in-tag>
       <mcast-prof>
        <type>McastProf</type>
        <id>
         <mcastprof>5</mcastprof>
        </id>
       </mcast-prof>
       <pon-cos>derived</pon-cos>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML create gateway service response on ONT RG Port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename="NTWK-WestE7" username="JDoe"
   sessionid="83">
   <ok/>
   <data>
    <top>
     <object>
      <type>EthSvc</type>
      <id>
       <ont>1</ont>
       <ontslot>8</ontslot>
       <ontethany>1</ontethany>
       <ethsvc name="Data1">1</ethsvc>
      </id>
      <admin>enabled</admin>
      <tag-action>
       <type>SvcTagAction</type>
       <id>
        <svctagaction name="@Ann_RG_1 1:1">158</svctagaction>
       </id>
      </tag-action>
      <bw-prof>
       <type>BwProf</type>
       <id>
        <bwprof name="@Ann">10</bwprof>
       </id>
      </bw-prof>
      <descr></descr>
      <out-tag>100</out-tag>
      <in-tag>none</in-tag>
      <mcast-prof>
       <type>McastProf</type>
       <id>
        <mcastprof name="@CMS_Video">5</mcastprof>
       </id>
```

```
      </mcast-prof>
      <pon-cos>derived</pon-cos>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating a gateway service request on a GPON ONT RG Port.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the FB ONT port using the number **9**. <ontethany> identifies the ONT port number. <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<descr>	Char(31)		Description of service

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (1 to 255).
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies an ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Element Tag	Data Type	Req'd ?	Description
<pon-cos>	see descr		Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

Update a gateway service on an RG port on a E7 GPON ONT

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) a gateway service on an E7 GPON ONT RG port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename=”NTWK-WestE7” username=”JDoe”
   sessionid=”83”>
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>EthSvc</type>
       <id>
        <ont>1</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
        . . .
```

Delete a gateway service on an RG port on a E7 GPON ONT

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename=”NTWK-WestE7” username=”JDoe”
   sessionid=”83”>
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="delete">
       <type>EthSvc</type>
       <id>
        <ont>1</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML delete response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="174" nodename=”NTWK-WestE7” username=”JDoe”
   sessionid=”83”>
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **delete** request for creating a gateway service request on a GPON ONT RG Port.

For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <ontslot> identifies the FB ONT port using the number **9**. <ontethany> identifies the ONT port number. <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service).

Suspending and Resuming E7 GPON ONT Services

The XML request defines the objects required to suspend or resume subscriber service on an E7 GPON ONT Ethernet, Voice, DS1, RF video, or Hot RF video port.

Update an E7 ONT port

Sample XML request for suspending service

The following shows an example of a suspend (merge) request for ONT Ethernet port 1.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="180" nodename="NTWK-e7_20" username="rootgod"
        sessionid="32">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntEthGe</type>
              <id>
                <ont>3</<ont>
                <ontslot>3</ontslot>
                <ontethge>1</onethge>
              </id>
              <admin>disabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a **merge** request for adding or updating the subscriber ID and user description on an E7 GPON port. For descriptions of common XML element tags, see "Common XML Element Tags" on page 123.

Element Tag	Data Type	Req'd ?	Description
<type>	See descr	Yes	<type> identifies the provisioning object type using on of the following case-sensitive expressions: • **OntDs1**—DS1 ports • **OntEthFe**—FE port • **OntEthGe**—GE port • **OntEthHpna**—HPNA port • **OntPots**—POTS ports • **OntRfAvo**—RF video
<id> <ont> <ontslot> [<ontethge>| <ontethhpna>| <ontethfe>|<ontpots>| <ontvideorf>|<ontvideorf>| <ontvideohotrf>|<ontfb>]	see descr	Yes	These element tags identify the port: <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—port type (one of the following): • **1**—RF Video • **2**—Hot RF Video • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port • **6**—Voice port • **7**—PWE3 port • **8**—RG port • **9**—Full Bridge (FB) port The port number is identified with one of the following prefixes: • <ontethge> — Gigabit Ethernet port • <ontethhpna> — HPNA Ethernet port • <ontethfe>— Fast Ethernet port • <ontpots> — Voice port • <ontvideorf>— RF Video • <ontvideohotrf> — Hot RF Video • <ontfb> - FB port.
<admin>	Char		Use one of the following: • **disabled**—suspend all traffic on the port • **enabled**—resume all traffic on the port

Sample XML update reply

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="2" nodename="NTWK-E7Test">
      <ok/>
      <data>
        <top>
          <object>
            <type>OntEthGe</type>
              <id>
                <ont>3</<ont>
                <ontslot>3</ontslot>
                <ontethge>1</onethge>
              </id>
            <admin>disabled</admin>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>lll
```

Updating a GPON ONT Registration ID

The XML request defines the objects required to update a GPON ONT registration ID.

Sample XML request for updating the registration ID on a GPON ONT

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="118" nodename="NTWK-412" username="rootgod" sessionid="6">
   <edit-config>
     <target>
      <running/>
     </target>
     <config>
       <top>
         <object operation="merge">
           <type>Ont</type>
           <id>
             <ont>1</ont>
           </id>
           <reg-id>1234567</reg-id>
         </object>
       </top>
     </config>
    </edit-config>
   </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="118" nodename="NTWK-412">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required to update a GPON ONT registration ID.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> </id> <reg-id>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT. <reg-id> ONT registration ID that is the RONTA identifier.

Query ONTs by Reg-ID

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
<soapenv:Body>
<rpc message-id="37" nodename="NTWK-412" username="rootgod" sessionid="33">
    <action>
        <action-type>show-ont</action-type>
            <action-args>
                <reg-id>111</reg-id>
            </action-args>
    </action>
</rpc>
</soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope>
  <soapenv:Body>
    <rpc-reply message-id="37" nodename="NTWK-412">
           <ok />
            <action-reply>
            <match>
           <get-config>
           <object>
           <type>Ont</type>
           <id>
            <ont>1</ont>
           </id>
           <admin>enabled</admin>
           <ontprof>
            <type>OntProf</type>
```

```
 <id>
  <ontprof name="836GE">148</ontprof>
 </id>
</ontprof>
 <serno>62074</serno>
 <reg-id>111</reg-id>
<subscr-id>Test hz5</subscr-id>
  <descr>Test hz35</descr>
  <linked-pon />
  <pwe3prof />
  <low-rx-opt-pwr-ne-thresh>-30.00</low-rx-opt-pwr-ne-thresh>
  <high-rx-opt-pwr-ne-thresh>-7.00</high-rx-opt-pwr-ne-thresh>
  <us-sdber-rate>5</us-sdber-rate>
  <low-rx-opt-pwr-fe-thresh>-30.00</low-rx-opt-pwr-fe-thresh>
  <high-rx-opt-pwr-fe-thresh>-7.00</high-rx-opt-pwr-fe-thresh>
  <low-tx-opt-pwr-thresh>-16.00</low-tx-opt-pwr-thresh>
  <high-tx-opt-pwr-thresh>10.00</high-tx-opt-pwr-thresh>
  <low-laser-bias-thresh>0</low-laser-bias-thresh>
  <high-laser-bias-thresh>75000</high-laser-bias-thresh>
  <low-line-pwr-feed-thresh>3000</low-line-pwr-feed-thresh>
  <high-line-pwr-feed-thresh>3590</high-line-pwr-feed-thresh>
  <low-ont-temp-thresh>-45</low-ont-temp-thresh>
      <high-ont-temp-thresh>120</high-ont-temp-thresh>
      <battery-present>true</battery-present>
     </object>
    </get-config>
     <get>
      <object>
       <type>Ont</type>
      <id>
       <ont>1</ont>
      </id>
      <op-stat>sys-disable</op-stat>
      <crit>0</crit>
      <maj>0</maj>
      <min>0</min>
      <warn>0</warn>
      <info>0</info>
      <derived-states>child-prov</derived-states>
       <model />
       <vendor />
       <clei />
       <ds-sdber-rate>0</ds-sdber-rate>
       <product-code>none</product-code>
       <mfg-serno />
       <uptime>0</uptime>
       <opt-sig-lvl>0.000</opt-sig-lvl>
       <tx-opt-lvl>0.000</tx-opt-lvl>
       <range-length>0</range-length>
       <fe-opt-lvl>0.000</fe-opt-lvl>
       <cur-ds-sdber-rate>0</cur-ds-sdber-rate>
       <cur-us-sdber-rate>0</cur-us-sdber-rate>
       <curr-sw-vers />
       <alt-sw-vers />
       <curr-committed>false</curr-committed>
```

```
                    <rg-config-file-vers />
                    <voip-config-file-vers />
                    <curr-cust-vers />
                    <alt-cust-vers />
                    <onu-mac>00:00:00:00:00:00</onu-mac>
                    <mta-mac>00:00:00:00:00:00</mta-mac>
                    <response-time>0</response-time>
                   </object>
                  </get>
                 </match>
                 <more />
                </action-reply>
               </rpc-reply>
        </soapenv:Body>
</soapenv:Envelope>
```

Replacing a GPON ONT

This section describes the XML requests that define the objects required to replace an E7 GPON ONT. The steps to replace a GPON ONT are as follows:

- First unlink the ONT from the provisioning record. After the ONT is unlinked, the provisioned ONT data is still on the E7, but the serial number (<serno0> is set to zero (0). This is described below in "Unlink the GPON ONT from the provisioning record" on page 240.
- Second, install the new GPON ONT and discover it by the E7.
- Third, link the discovered ONT to the existing ONT provisioning by editing the serial number in an existing provisioning record to match the serial number of the discovered ONT. This is described below in "Link a newly discovered ONT to existing ONT provisioning" on page 242.

Unlink the GPON ONT from the provisioning record

Sample XML request to unlink the ONT

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="37" nodename="NTWK-GLASS" timeout="35000"
username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>Ont</type>
       <id>
        <ont>1</ont>
       </id>
       <serno>0</serno>
       <linked-pon></linked-pon>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to unlink the ONT

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="37" nodename="NTWK-GLASS">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required to unlink a GPON ONT from the provisioning record.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> </id>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT.
<serno>	see descr	Yes	Set the serial number field to zero (0).

Link a newly discovered ONT to existing ONT provisioning

Sample XML request to link a newly discovered ONT to existing ONT provisioning

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="37" nodename="NTWK-GLASS" timeout="35000"
username="rootgod" sessionid="10">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>Ont</type>
       <id>
        <ont>1</ont>
       </id>
       <admin>enabled</admin>
       <serno>62074</serno>
       <reg-id></reg-id>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="37" nodename="NTWK-GLASS">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required to link a newly discovered ONT to existing ONT provisioning.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> </id>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT.
<admin>	see descr	Yes	Operational status: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
<serno>	see descr	Yes	<serno> Hexadecimal representation of the newly discovered ONT's serial number
<reg-id>	see descr	Yes	Leave this field empty.

Applying a Template to a GPON ONT

Sample XML FB-1 Interface Request

```
<rpc message-id="175" nodename="" timeout="35000">
       <edit-config>
              <target>
                     <running/>
              </target>
              <config>
                     <top>
                            <object operation="merge">
                                   <type>OntEthGe</type>
                                   <id>
                                          <ont>1</ont>
                                          <ontslot>3</ontslot>
                                          <ontethge>2</ontethge>
                                   </id>
                                   <intf>
                                          <type>OntFb</type>
                                          <id>
                                                 <ont>1</ont>
```

```
                                                <ontslot>9</ontslot>
                                                <ontfb>1</ontfb>
                                        </id>
                                </intf>
                        </object>
                </top>
        </config>
    </edit-config>
</rpc>
```

Sample XML RG-1 Interface Request

```
<rpc message-id="175" nodename="" timeout="35000">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <top>
                <object operation="merge">
                    <type>OntEthGe</type>
                    <id>
                        <ont>1</ont>
                        <ontslot>3</ontslot>
                        <ontethge>2</ontethge>
                    </id>
                    <intf>
                        <type>OntRg</type>
                        <id>
                            <ont>1</ont>
                            <ontslot>8</ontslot>
                            <ontrg>1</ontrg>
                        </id>
                    </intf>
                </object>
            </top>
        </config>
    </edit-config>
</rpc>
```

Retrieving Performance Data for GPON ONTs on the E7-20 Platform

This section describes the XML requests used for retrieving performance data for GPON ONTs on the E7-20 platform.

This section covers the following topics:

Get ONT Eth Port Admin State, Speed and Duplex

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="202" nodename="" timeout="35000">
   <get-config>
    <source>
     <running />
    </source>
    <filter type="subtree">
     <top>
      <object>
       <type>OntEthGe</type>
       <id>
        <ont>836</ont>
        <ontslot>3</ontslot>
        <ontethge>1</ontethge>
       </id>
      </object>
     </top>
    </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000765" message-id="202" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntEthGe</type>
      <id>
       <ont>836</ont>
       <ontslot>3</ontslot>
       <ontethge>1</ontethge>
      </id>
      <admin>enabled-no-alarms</admin>
      <subscr-id>707-123-2345</subscr-id>
      <descr>Bedrock Apts.</descr>
      <speed>auto</speed>
      <gos>
       <type>OntEthPortGos</type>
       <id>
        <ontethportgos>1</ontethportgos>
       </id>
      </gos>
```

```
      <duplex>full</duplex>
      <sec>
       <type>EthSecProf</type>
       <id>
        <ethsecprof name="@2222">16</ethsecprof>
       </id>
      </sec>
      <disable-on-batt>true</disable-on-batt>
      <link-oam-events>false</link-oam-events>
      <accept-link-oam-loopbacks>false</accept-link-oam-loopbacks>
      <intf></intf>
      <pbit-map>
       <type>DscpMap</type>
       <id>
        <dscpmap name="access">1</dscpmap>
       </id>
      </pbit-map>
      <dhcp-limit-override>none</dhcp-limit-override>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT Eth Port DHCP Leases

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="211" nodename="NTWK-E7-20-212" timeout="35000"
   username="rootgod" sessionid="142">
   <action>
    <action-type>show-dhcp-leases</action-type>
    <action-args>
     <object>
      <type>OntEthGe</type>
      <id>
       <ont>844</ont>
       <ontslot>8</ontslot>
       <ontrg>1</ontrg>
      </id>
     </object>
    </action-args>
   </action>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply message-id="211" nodename="NTWK-E7-20-212">
   <ok />
   <action-reply>
    <entry>
     <outer-vlan>600</outer-vlan>
     <inner-vlan>none</inner-vlan>
     <mac>00:03:e6:68:19:e6</mac>
     <port>
      <type>GponPort</type>
      <id>
       <shelf>1</shelf>
       <card>20</card>
       <gponport>2</gponport>
      </id>
     </port>
     <sub-port>
      <type>OntRg</type>
      <id>
       <ont>844</ont>
       <ontslot>8</ontslot>
       <ontrg>1</ontrg>
      </id>
     </sub-port>
     <ip>10.26.136.200</ip>
     <netmask>255.255.255.0</netmask>
     <gw>10.26.136.1</gw>
     <server>10.20.208.2</server>
     <is-static>false</is-static>
     <expiry-time>1422487209</expiry-time>
    </entry>
    <entry>
     <outer-vlan>600</outer-vlan>
     <inner-vlan>none</inner-vlan>
     <mac>00:02:02:48:95:21</mac>
     <port>
      <type>GponPort</type>
      <id>
       <shelf>1</shelf>
       <card>20</card>
       <gponport>2</gponport>
      </id>
     </port>
     <sub-port>
      <type>OntRg</type>
      <id>
       <ont>844</ont>
       <ontslot>8</ontslot>
       <ontrg>1</ontrg>
      </id>
     </sub-port>
     <ip>10.26.136.236</ip>
```

```
 <netmask>255.255.255.0</netmask>
 <gw>10.26.136.1</gw>
 <server>10.20.208.2</server>
 <is-static>false</is-static>
 <expiry-time>1422487205</expiry-time>
</entry>
<entry>
 <outer-vlan>600</outer-vlan>
 <inner-vlan>none</inner-vlan>
 <mac>00:02:02:48:c7:42</mac>
 <port>
  <type>GponPort</type>
  <id>
   <shelf>1</shelf>
   <card>20</card>
   <gponport>2</gponport>
  </id>
 </port>
 <sub-port>
  <type>OntRg</type>
  <id>
   <ont>844</ont>
   <ontslot>8</ontslot>
   <ontrg>1</ontrg>
  </id>
 </sub-port>
 <ip>10.26.136.237</ip>
 <netmask>255.255.255.0</netmask>
 <gw>10.26.136.1</gw>
 <server>10.20.208.2</server>
 <is-static>false</is-static>
 <expiry-time>1422487141</expiry-time>
</entry>
<entry>
 <outer-vlan>601</outer-vlan>
 <inner-vlan>none</inner-vlan>
 <mac>00:06:31:b4:e9:f9</mac>
 <port>
  <type>GponPort</type>
  <id>
   <shelf>1</shelf>
   <card>20</card>
   <gponport>2</gponport>
  </id>
 </port>
 <sub-port>
  <type>OntRg</type>
  <id>
   <ont>844</ont>
   <ontslot>8</ontslot>
   <ontrg>1</ontrg>
  </id>
 </sub-port>
 <ip>10.26.137.22</ip>
 <netmask>255.255.255.0</netmask>
```

```
     <gw>10.26.137.1</gw>
     <server>10.21.81.1</server>
     <is-static>false</is-static>
     <expiry-time>1422555418</expiry-time>
    </entry>
   </action-reply>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT Eth Port and Operation Status Performance Data

Sample XML request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="203" nodename="" timeout="35000">
   <get>
    <filter type="subtree">
     <top>
      <object>
       <type>OntEthGe</type>
       <id>
        <ont>836</ont>
        <ontslot>3</ontslot>
        <ontethge>1</ontethge>
       </id>
      </object>
     </top>
    </filter>
   </get>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="001553" message-id="203" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntEthGe</type>
      <id>
       <ont>836</ont>
       <ontslot>3</ontslot>
       <ontethge>1</ontethge>
      </id>
      <op-stat>sys-disable</op-stat>
      <crit>0</crit>
```

```
        <maj>1</maj>
        <min>0</min>
        <warn>0</warn>
        <info>0</info>
        <derived-states>suppr</derived-states>
        <rate>10</rate>
        <status>down</status>
        <actual-duplex>half</actual-duplex>
        <in-discards>0</in-discards>
        <in-errors>0</in-errors>
        <in-octet>0</in-octet>
        <in-ucast>0</in-ucast>
        <in-mcast>0</in-mcast>
        <in-bcast>0</in-bcast>
        <out-octet>0</out-octet>
        <out-ucast>0</out-ucast>
        <out-mcast>0</out-mcast>
        <out-bcast>0</out-bcast>
        <in-rate>0</in-rate>
        <in-max-rate>0</in-max-rate>
        <out-rate>0</out-rate>
        <out-max-rate>0</out-max-rate>
        <power-status>ac-up</power-status>
        <mac>00:06:31:54:06:16</mac>
        <rfc2544-loopback>false</rfc2544-loopback>
        <rfc2544-vlan>0</rfc2544-vlan>
        <sfp-status>ont-does-not-support-sfp</sfp-status>
        <sfp-type>unknown</sfp-type>
        <sfp-conn>unknown</sfp-conn>
        <sfp-encoding>unknown</sfp-encoding>
        <sfp-bitrate>0</sfp-bitrate>
        <sfp-bitratemax>0</sfp-bitratemax>
        <sfp-vendname></sfp-vendname>
        <sfp-vendpartno></sfp-vendpartno>
        <sfp-vendrev></sfp-vendrev>
        <sfp-vendserno></sfp-vendserno>
        <sfp-temp>0.0</sfp-temp>
        <sfp-tx-bias>0.000</sfp-tx-bias>
        <sfp-tx-power>-999.999</sfp-tx-power>
        <sfp-rx-power>-999.999</sfp-rx-power>
        <sfp-voltage>0.00</sfp-voltage>
        <sfp-line-length>0</sfp-line-length>
        <sfp-wavelength>0.00</sfp-wavelength>
        <clei></clei>
        <link-oam-status>down</link-oam-status>
        <mtu>2000</mtu>
        <link-oam-loopback>none</link-oam-loopback>
       </object>
      </top>
     </data>
    </rpc-reply>
   </soapenv:Body>
  </soapenv:Envelope>
```

Get ONT Eth Service Bandwidth Profile and Tags

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="146" nodename="" timeout="35000">
   <get-config>
    <source>
     <running />
    </source>
    <filter type="subtree">
     <top>
      <object>
       <type>OntFb</type>
       <id>
        <ont>836</ont>
        <ontslot>9</ontslot>
        <ontfb>1</ontfb>
       </id>
       <children>
        <type>EthSvc</type>
        <attr-list>admin descr tag-action bw-prof out-tag in-tag
         mcast-prof pon-cos us-cir-override us-pir-override
         ds-pir-override</attr-list>
       </children>
      </object>
     </top>
    </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000870" message-id="146" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntFb</type>
      <id>
       <ont>836</ont>
       <ontslot>9</ontslot>
       <ontfb>1</ontfb>
      </id>
      <children>
       <child>
        <type>EthSvc</type>
        <id>
```

```
          <ont>836</ont>
          <ontslot>9</ontslot>
          <ontethany>1</ontethany>
          <ethsvc name="Video1">9</ethsvc>
         </id>
         <admin>enabled</admin>
         <descr></descr>
         <tag-action>
          <type>SvcTagAction</type>
          <id>
           <svctagaction name="change_tag">17</svctagaction>
          </id>
         </tag-action>
         <bw-prof>
          <type>BwProf</type>
          <id>
           <bwprof name="JamesbwProfile">93</bwprof>
          </id>
         </bw-prof>
         <out-tag>3</out-tag>
         <in-tag>none</in-tag>
         <mcast-prof>
          <type>McastProf</type>
          <id>
           <mcastprof name="JimMCast">18</mcastprof>
          </id>
         </mcast-prof>
         <pon-cos>derived</pon-cos>
         <us-cir-override>none</us-cir-override>
         <us-pir-override>none</us-pir-override>
         <ds-pir-override>none</ds-pir-override>
        </child>
       </children>
      </object>
     </top>
    </data>
   </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

Get ONT Model and Operation Status

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="160" nodename="" timeout="35000">
   <get>
    <filter type="subtree">
     <top>
      <object>
       <type>Ont</type>
       <id>
        <ont>1001</ont>
       </id>
      </object>
     </top>
    </filter>
   </get>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

Note: For E7 R2.3.10 and higher, this command will return ONU MAC and MTA MAC data.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="001028" message-id="160" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>Ont</type>
      <id>
       <ont>1001</ont>
      </id>
      <op-stat>enable</op-stat>
      <crit>0</crit>
      <maj>0</maj>
      <min>0</min>
      <warn>0</warn>
      <info>0</info>
      <derived-states>child-prov present</derived-states>
      <model>836GE</model>
      <vendor>CXNK</vendor>
      <clei>BVMBOKAIDU</clei>
      <ds-sdber-rate>5</ds-sdber-rate>
      <product-code>S8</product-code>
      <uptime>1213347.000</uptime>
```

```
      <opt-sig-lvl>na</opt-sig-lvl>
      <tx-opt-lvl>2.478</tx-opt-lvl>
      <range-length>90</range-length>
      <fe-opt-lvl>-12.000</fe-opt-lvl>
      <cur-ds-sdber-rate>9</cur-ds-sdber-rate>
      <cur-us-sdber-rate>8</cur-us-sdber-rate>
      <curr-sw-vers>10.6.40.4</curr-sw-vers>
      <alt-sw-vers>10.6.30.2</alt-sw-vers>
      <curr-committed>true</curr-committed>
      <rg-config-file-vers>
      </rg-config-file-vers>
      <voip-config-file-vers>
      </voip-config-file-vers>
      <curr-cust-vers></curr-cust-vers>
      <alt-cust-vers></alt-cust-vers>
      <onu-mac>00:06:31:47:8e:b7</onu-mac>
      <mta-mac>00:06:31:54:06:13</mta-mac>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT PM Stats

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="88" nodename="" timeout="35000">
   <action>
    <action-type>show-ont-pm</action-type>
    <action-args>
     <object>
      <type>Ont</type>
      <id>
       <ont>836</ont>
      </id>
     </object>
     <bin-type>total</bin-type>
     <start-bin>1</start-bin>
     <count>1</count>
    </action-args>
   </action>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
```

```
 <soapenv:Body>
  <rpc-reply length="000522" message-id="88" nodename=""
   timeout="35000">
   <ok />
   <action-reply>
    <types>bip-err-up miss-burst-up gem-hec-err-up bip-err-down
     bip-err-sec-up bip-sev-err-sec-up bip-unavail-sec-up
     missed-burst-sec bip-err-sec-down bip-sev-err-sec-down
     bip-unavail-sec-down</types>
    <bin>
     <num>1</num>
     <start>1412816427</start>
     <sec>1228760</sec>
     <valid>remote-arrived</valid>
     <val>1 0 0 0 1 0 0 0 0 0 0</val>
     <tca>false false false false false false false false false false
      false</tca>
    </bin>
    <max />
   </action-reply>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT POTS Port Administrative State

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="171" nodename="" timeout="35000">
   <get-config>
    <source>
     <running />
    </source>
    <filter type="subtree">
     <top>
      <object>
       <type>OntPots</type>
       <id>
        <ont>836</ont>
        <ontslot>6</ontslot>
        <ontpots>1</ontpots>
       </id>
      </object>
     </top>
    </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000471" message-id="171" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntPots</type>
      <id>
       <ont>836</ont>
       <ontslot>6</ontslot>
       <ontpots>1</ontpots>
      </id>
      <admin>enabled</admin>
      <subscr-id></subscr-id>
      <descr></descr>
      <impedance>600-ohm</impedance>
      <signal-type>loop-start</signal-type>
      <system-tx-loss>gr909</system-tx-loss>
      <system-rx-loss>gr909</system-rx-loss>
      <tx-gain-2db>0.0</tx-gain-2db>
      <rx-gain-2db>0.0</rx-gain-2db>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT POTS Port Operational Status

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="172" nodename="" timeout="35000">
   <get>
    <filter type="subtree">
     <top>
      <object>
       <type>OntPots</type>
       <id>
        <ont>836</ont>
        <ontslot>6</ontslot>
        <ontpots>1</ontpots>
       </id>
      </object>
     </top>
    </filter>
   </get>
  </rpc>
```

```
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000418" message-id="172" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntPots</type>
      <id>
       <ont>836</ont>
       <ontslot>6</ontslot>
       <ontpots>1</ontpots>
      </id>
      <op-stat>enable</op-stat>
      <crit>0</crit>
      <maj>0</maj>
      <min>0</min>
      <warn>0</warn>
      <info>0</info>
      <derived-states>default-prov</derived-states>
      <power-status>ac-up</power-status>
      <tx-gain>0</tx-gain>
      <rx-gain>0</rx-gain>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT RF Video Port Administrative State

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="214" nodename="" timeout="35000">
   <get-config>
    <source>
     <running />
    </source>
    <filter type="subtree">
     <top>
      <object>
       <type>OntVideoRf</type>
       <id>
        <ont>5</ont>
```

```
        <ontslot>1</ontslot>
        <ontvideorf>1</ontvideorf>
       </id>
      </object>
     </top>
    </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000315" message-id="214" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntVideoRf</type>
      <id>
       <ont>5</ont>
       <ontslot>1</ontslot>
       <ontvideorf>1</ontvideorf>
      </id>
      <admin>disabled</admin>
      <subscr-id></subscr-id>
      <descr></descr>
      <disable-on-batt>true</disable-on-batt>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT RF Video Port Operational Status

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="215" nodename="" timeout="35000">
   <get>
    <filter type="subtree">
     <top>
      <object>
       <type>OntVideoRf</type>
       <id>
        <ont>5</ont>
        <ontslot>1</ontslot>
```

```
        <ontvideorf>1</ontvideorf>
       </id>
      </object>
     </top>
    </filter>
   </get>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000391" message-id="215" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>OntVideoRf</type>
      <id>
       <ont>5</ont>
       <ontslot>1</ontslot>
       <ontvideorf>1</ontvideorf>
      </id>
      <op-stat>user-disable</op-stat>
      <crit>0</crit>
      <maj>0</maj>
      <min>0</min>
      <warn>0</warn>
      <info>0</info>
      <derived-states>default-prov</derived-states>
      <power-status>ac-up</power-status>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Get ONT Serial Number

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="159" nodename="" timeout="35000">
   <get-config>
    <source>
     <running />
    </source>
    <filter type="subtree">
     <top>
      <object>
```

```
       <type>Ont</type>
       <id>
        <ont>1001</ont>
       </id>
      </object>
     </top>
    </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000624" message-id="159" nodename=""
   timeout="35000">
   <data>
    <top>
     <object>
      <type>Ont</type>
      <id>
       <ont>1001</ont>
      </id>
      <admin>enabled</admin>
      <ontprof>
       <type>OntProf</type>
       <id>
        <ontprof name="836GE">148</ontprof>
       </id>
      </ontprof>
      <serno>8e5cb</serno>
      <reg-id></reg-id>
      <subscr-id></subscr-id>
      <descr></descr>
      <linked-pon>
       <type>GponPort</type>
       <id>
        <shelf>1</shelf>
        <card>13</card>
        <gponport>3</gponport>
       </id>
      </linked-pon>
      <pwe3prof></pwe3prof>
      <low-rx-opt-pwr-ne-thresh>-30.00</low-rx-opt-pwr-ne-thresh>
      <high-rx-opt-pwr-ne-thresh>-7.00</high-rx-opt-pwr-ne-thresh>
      <us-sdber-rate>5</us-sdber-rate>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Setting Up a Gateway Service on an 844G GigaCenter

This section describes the XML requests used for retrieving performance data for GPON ONTs on the E7-20 platform.

This section covers the following topics:

Setting up the Upstream Rate Override, Upstream Peak Rate Override, and Downstream Peak Rate

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply length="000624" message-id="159" nodename=""
   timeout="35000">
 <edit-config>
  <target>
   <running />
  </target>
  <config>
   <top>
    <object operation="create"get-config="true">
     <type>EthSvc</type>
     <id>
      <ont>3</ont>
      <ontslot>8</ontslot>
      <ontethany>1</ontethany>
      <ethsvc>2</ethsvc>
     </id>
     <admin>enabled</admin>
     <descr>
     </descr>
     <tag-action>
      <type>SvcTagAction</type>
      <id>
       <svctagaction>1</svctagaction>
      </id>
     </tag-action>
     <bw-prof>
      <type>BwProf</type>
      <id>
       <bwprof>120</bwprof>
      </id>
     </bw-prof>
```

```
     <out-tag>none</out-tag>
     <in-tag>none</in-tag>
     <mcast-prof>
     </mcast-prof>
     <pon-cos>derived</pon-cos>
     <us-cir-override>none</us-cir-override>
     <us-pir-override>50m</us-pir-override>
     <ds-pir-override>none</ds-pir-override>
    </object>
   </top>
  </config>
 </edit-config>
</rpc>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags" on page 123.

The following table lists the XML element tags required to link a newly discovered ONT to existing ONT provisioning.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`  <ont>`
`</id>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**). The nodename attribute value in the <rpc> tag and the <ont> tag value identify the GPON ONT.

`<us-cir-override>`			
`<us-pir-override>`			
`<ds-pir-override>`			

AE ONT Service Activation and Querying

This section presents the following topics:

Guidelines for AE ONT Provisioning

When using the examples and explanations in this guide, keep in mind the following:

- The session ID attribute in the <rpc> tag for E7 and AE ONT requests contain a lowercase "i" (sessionid); for other Calix network elements the attribute has an uppercase "I" (sessionId).
- AE ONT suspend and resume service requests are not yet supported.
- Using the XML NBI for AE ONT, you can only activate one service type per XML request. For more than one service on the same ONT, you must send separate XML requests.
- XML service activation requests override any previously provisioned service parameters.
- After service provisioning, AE ONT configuration files must be saved and the AE ONT reset. XML examples are provided at the end of this chapter.
- For non-required element tags, if the tag is not included in the XML request, the default value of the parameter overrides the currently provisioned value. Calix recommends including these element tags in XML requests even if the value is not changing.
- Special characters used in the management interface, when returned within element tags, are escaped. Refer to the following table for examples.

Special character	After escaping
<	<
>	>
&	&
"	"

  For example, a global multicast profile name defined as @Video&200 is returned in the XML replies as mcastprof name="@Video&200". Follow company policies and procedures when assigning names in the management interface.
- The AE ONT referenced by the XML attributes must support the requested service. For specific information refer to the Calix AE ONT documentation.
- CMS releases prior to 11.2 used the fields "**unknown**", "**enabled**" and "**disabled**" for the AE ONT port <admin> element. When CMS was upgraded to CMS 11.2, the attribute "**enabled**" was automatically changed to "**enabled-no-alarms**" for AE ONT port settings. This is the default setting for AE ONTs in CMS releases 11.2 and higher. The attribute "**ienabled**" is used instead of "enabled" for AE ONT port <admin> settings.

Common XML Element Tags for Edit-Config Requests

This topic covers the following topics:

- "Common element tags for AE ONT edit-configuration XML Requests" (see below).
- "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Element tags for read (get-configuration) and Save Configuration/Reset AE ONT XML requests are unique. See "XML Read (get-config) Requests" on page 271 and "Saving and Resetting AE ONTs" on page 359, respectively.

Common element tags for AE ONT edit-configuration XML Requests

Each AE ONT XML service activation request contains a SOAP envelope and tags, as shown in the following example excerpt.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="37" nodeame="AeCMSNetwork" username="JDoe"
        sessionid="14">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              . . .
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML the common element tags in AE ONT **create**, **merge**, and **delete** requests.

Element Tag	Data Type	Req'd ?	Description
<rpc ...>	see below	Yes	This tag is nested under the <soapenv:Body> tag, and contains the attributes for authenticating the XML request, as described in the next four rows.

Element Tag	Data Type	Req'd ?	Description
message-id=	Positive Integer: 2^31-1		A unique number identifying the request, and used to match the XML reply with the request.
nodename=	Char	Yes	Use the following case-sensitive expression for all AE ONT requests: "AeCMSNetwork" **Note:** The <aeontid> tag specified in the request identifies the AE ONT.
username=	Char	Yes	The name of the user currently logged in to the XML NBI, enclosed in quotes. Example: "**JDoe**"
sessionid=	Int	Yes	Use the session ID returned after logging in to the CMS NBI (1 to 150). **Note:** The session ID attribute in the <rpc> tag for E7 and AE ONT CMS requests contain a lowercase "i" (sessionid); for other Calix network elements the attribute has an uppercase "I" (sessionId).
<edit-config>	N/A	Yes	This tag is nested under <rpc> tag and identifies the request as an edit-configuration type. For read requests, <get-config> is used instead. See "XML Read (get-config) Requests" on page 271.
<object ...>	Char	Yes	This tag is nested under the <top> tag and contains the attributes for defining the XML request, as described in the next two rows.
operation=	Char	Yes	Identifies the requested action: • "**create**" • "**merge**" • "**delete**"
get-config=			This attribute is not supported in CMS R12.0.

The sections below in this chapter describe specific element tags for each request type nested under the <object> tag.

Common element tags for AE ONT edit-configuration XML replies

Each AE ONT XML reply contains a SOAP envelope and tags, as shown in the following example excerpts.

The following example excerpt shows the results of a successful provisioning request.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="37" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows the results of an unsuccessful provisioning request.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="00359" message-id="100" nodename="AeCMSNetwork">
      <rpc-error>
        <error-type>application</error-type>
        <error-tag>operation-failed</error-tag>
        <error-severity>error</error-severity>
        <error-app-tag>inconsistent-values</error-app-tag>
        <error-message xml:lang="en">tag-action used by ip-host must specify
         expedited COS queue.</error-message>
     </rpc-error>
    </rpc-reply>
   </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML common element tags in AE ONT replies.

Element Tag	Data Type	Description
`<rpc-reply ...>`	see descr	This tag is nested under <soapenv:Body> tag and identifies the message ID and AE ONT, as described in the next two rows.
`message-id=`	Positive Integer: 2^31-1	The message ID, as provided in the XML request, enclosed in quotes.
`nodename=`	Char	The following expression is returned in all AE ONT replies: "AeCMSNetwork"
Element tag for successful provisioning: `<ok/>`	Tag only	This tag is nested under the <rpc-reply> tag and indicates the request was successful.

Element Tag	Data Type	Description

Additional element tags for error conditions:
`<error-type>`
`<error-tag>`
`<error-severity>`
`<error-app-tag>`
`<error-message>`
	Char	These tags are nested under the <rpc-reply> tag and indicate the details of the error condition, including the error type, name, severity, application, and message.

XML Read (get-config) Requests

You can retrieve provisioning information from an AE ONT using these two methods:

- Query services on an AE ONT by service type (see below)
- Query a specific service on an AE ONT (“<bwprof name="">” on page 278)

Common element tags for get-config requests

Element Tag	Data Type	Req’d ?	Description
<rpc ...>		Yes	This tag is nested under the <soapenv:Body> tag and contains the attributes for authenticating the XML request, as described in the next four rows.
message-id=	Positive Integer: 2^31-1		A unique number identifying the request, enclosed in quotes, that is used to match the XML reply with the request.
nodename=	Char	Yes	Use the following case-sensitive expression for all AE ONT requests: "AeCMSNetwork" **Note:** The <aeontid> tag specified in the request identifies the AE ONT.
username=	Char	Yes	The name of the user currently logged in to the CMS XML NBI, enclosed in quotes.
sessionid=	Int	Yes	Use the session ID returned after logging in to the CMS XML NBI, enclosed in quotes. **Note:** The session ID attribute for E7 and AE ONT CMS requests contain a lowercase “i” (sessionid); for other Calix network elements the attribute has an uppercase “I” (sessionId).
<get-config>	N/A	Yes	This tag is nested under the <rpc> tag, and identifies the request as a get-configuration type.
<filter type="subtree">	see descr	Yes	Identifies the filter or scope of the request. Use the expression to the left in all cases.

Sample XML get-config request for services on an AE ONT by service type

The following example shows a read request for service on a GE port.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="118" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <get-config>
        <source>
          <running />
        </source>
        <filter type="subtree">
          <top>
            <object>
              <type>Ont</type>
              <id>
                <aeontid>CXNK00061675</aeontid>
              </id>
              <children>
                <type>EthSvc</type>
                <attr-list>admin descr tag-action bw-prof out-tag
                   in-tag mcast-prof
                </attr-list>
              </children>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags for get-config requests for multiple services.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the provisioning object type using the following case-sensitive expression: **Ont**
<id> <aeontid>	see descr	Yes	For ONT Ethernet ports, these element tags identify the ONT, slot, and port: <aeontid>—registration ID or FSAN serial number of the AE ONT.

Element Tag	Data Type	Req'd ?	Description
<children> <type> <attr-list>	see descr	Yes Yes No	<type> identifies the service type on the AE ONT using one of the following case-sensitive expressions: • **EthSvc**—Ethernet data and video service • **H248GwSvc**—H.248 VoIP service • **MgcpGwSvc**—MGCP VoIP service • **Pwe3Svc**—PWE3 DS1 service • **SipSvc**—SIP VoIP service • **TdmGwSvc**—TDM gateway VoIP service <attr-list> includes the configuration attributes to return. If no attributes are specified in the element tag, all attributes are returned. For a list of attributes, refer to the XML example and XML element tag tables in the following topics: • "Create data service on AE ONTs" on page 290 • "Create video service on AE ONTs" on page 299 • "Create SIP VoIP service on AE ONTs" on page 329 • "Create TDM Gateway VoIP service on AE ONTs" on page 335 • "Create H.248 VoIP service on AE ONTs" on page 340 • "Create MGCP VoIP service on AE ONTs" on page 345 • "Create PWE3 service on AE ONTs" on page 353

Sample XML get-config reply (for multiple AE ONT services)

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="30" nodename="AeCMSNetwork">
      <data>
        <top>
          <object>
            <type>Ont</type>
            <id>
              <aeontid>CXNK00061675</aeontid>
            </id>
            <children>
              <child>
                <type>EthSvc</type>
```

```
                <id>
                  <aeontid name="">766018</ont>
                  <ontslot>3</ontslot>
                  <ontethany>1</ontethany>
                  <ethsvc name="Data1">1</ethsvc>
                </id>
                <admin>enabled</admin>
                <descr>OAM-EP-1</descr>
                <tag-action>
                  <type>SvcTagAction</type>
                  <id><svctagaction name="DataTag1">1</svctagaction></id>
                </tag-action>
                <bw-prof>
                  <type>BwProf</type>
                    <id>
                      <bwprof name="af-10M">4</bwprof>
                    </id>
                  </bw-prof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof></mcast-prof>
              </child>
              <child>
                <type>EthSvc</type>
                <id>
                  <aeontid name="">766018</ont>
                  <ontslot>3</ontslot>
                  <ontethany>4</ontethany>
                  <ethsvc name="Data1">1</ethsvc>
                </id>
                <admin>enabled</admin>
                <descr></descr>
                <tag-action>
                  <type>SvcTagAction</type>
                  <id>
                    <svctagaction name="CT-Same_UP-MEP">10</svctagaction>
                  </id>
                </tag-action>
                <bw-prof>
                  <type>BwProf</type>
                  <id>
                    <bwprof name="be">10</bwprof>
                  </id>
                </bw-prof>
                <out-tag>none</out-tag>
                <in-tag>none</in-tag>
                <mcast-prof></mcast-prof>
              </child>
            </children>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

If provisioned, the attribute name value is returned in the following element tags:

- <aeontid name="">
- Depending on the type of service: <ethsvc name="">, <sipsvc name="">, <tdmgwsvc name="">, <h248gwsvc name="">, <mgcpgwsvc name="">, or <pwe3svc name="">
- <svctagaction name="">
- <bwprof name="">

Sample XML get-config request for a specific object or service

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="118" nodename="AeCMSNetwork" username="JDoe"
         sessionid="37">
      <get-config>
        <source><running/></source>
        <filter type="subtree">
          <top>
            <object>
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00061675</aeontid>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </filter>
      </get-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the specific XML element tags used for get-config requests for a specific service. "Common element tags for AE ONT edit-configuration XML Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<object> <type>	see descr	Yes	Identifies the service or object using one of the following case-sensitive expressions: Services • **EthSvc**—Ethernet data or video service • **SipSvc**—SIP service • **TdmGwSvc**—TDM Gateway service • **H248GwSvc**—H.248 service • **MgcpGwSvc**—MGCP service • **Pwe3Svc**—PWE3 service IP Host (when querying a single IP host) • **OntIpHost**—IP host ONT and ONT ports • **Ont**—ONT object, including queries for multiple IP hosts • **OntDs1**—DS1 ports • **OntEthFe**—FE port • **OntEthGe**—GE port • **OntEthHpna**—HPNA port • **OntPots**—POTS ports
For data and video services (<EthSvc> type): <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	These element tags identify the port: <aeontid>—identifies the ONT using the registration ID or FSAN serial number of the AE ONT. <ontslot>—port type (one of the following): • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany>—ONT Ethernet port number (**1** to **8**). <ethsvc>—data or video service number (**1** to **12**).

Element Tag	Data Type	Req'd ?	Description

For SIP or TDM Gateway VoIP service:
`<id>`
`  <aeontid>`
`  <ontslot>`
`  <ontpots>`
`  [<sipsvc>|<tdmgwsvc>]`
	see descr	Yes	These element tags identify the port:
			`<aeontid>`—identifies the ONT using the registration ID or FSAN serial number of the AE ONT.
			`<ontslot>`—port type (**6**).
			`<ontpots>`—ONT voice port number (**1** to **8**).
			`<sipsvc>` or `<tdmgwsvc>`—voice service number (**1**).

For H.248 or MGCP VoIP service:
`<id>`
`  <aeontid>`
`  <ontslot>`
`  <ontpots>`
`  [<h248svc>|<mgcpsvc>]`
	see descr	Yes	These element tags identify the port:
			`<aeontid>`—identifies the ONT using the registration ID or FSAN serial number of the AE ONT.
			`<ontslot>`—port type (**6**).
			`<ontpots>`—ONT voice port number (**1** to **8**).
			`<h248svc>` or `<mgcpsvc>`—voice service number (**1** to **20**).

For PWE3 service:
`<id>`
`  <aeontid>`
`  <ontslot>`
`  <ontds1>`
`  <pwe3svc>`
	see descr	Yes	These element tags identify the port:
			`<aeontid>`—identifies the ONT using the registration ID or FSAN serial number of the AE ONT.
			`<ontslot>`—port type (**7**).
			`<ontds1>`—ONT Ethernet port number (**1** to **8**).
			`<pwe3svc>`—service number (**1**).

For querying multiple IP hosts:
`<id>`
`  <aeontid>`
`<children>`
`  <type>`
	see descr	Yes	These element tags identify the object:
			`<aeontid>`—identifies the ONT using the registration ID or FSAN serial number of the AE ONT.
			`<type>`—identifies the object (**OntIpHost**).

For ONT objects:
`<id>`
`  <aeontid>`
	see descr	Yes	`<aeontid>`—identifies the ONT using the registration ID or FSAN serial number of the AE ONT.

Sample XML get-config reply (for a single AE ONT service)

Note: The following example shows an XML reply for data service.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply length="000545" message-id="147" nodename="AeCMSNetwork”>
      <data>
        <top>
          <object>
            <type>EthSvc</type>
            <id>
              <aeontid name="">766029</aeontid>
              <ontslot>3</ontslot>
              <ontethany>1</ontethany>
              <ethsvc name="Data1">1</ethsvc>
            </id>
            <admin>enabled</admin>
            <tag-action>
              <type>SvcTagAction</type>
              <id>
                <svctagaction name=”Test_data_svc”>3</svctagaction>
              </id>
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof name=”Test_bw_profile”>1</bwprof>
              </id>
            </bwprof>
            <descr/>
            <out-tag>none</out-tag>
            <in-tag>none</in-tag>
          </object>
        </top>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

If provisioned, the attribute name value is returned in the following element tags:

- <ont name="">
- Depending on the service type: <ethsvc name="">, <sipsvc name="">, <tdmgwsvc name="">, <h248gwsvc name="">, <mgcpgwsvc name="">, or <pwe3svc name="">
- <svctagaction name="">
- <bwprof name="">

Creating and Deleting AE ONT ONTs

The XML request defines the objects required to create and delete AE ONTs.

This section contains the following topics:

Global ONT Profile IDs

The following global profile IDs can be used when creating AE ONTs:

Profile ID	Model
619	710GX
618	711GX
617	712GX
616	714GX
615	720GX
614	721GX
613	722GX
612	724GX
611	725GX
610	760GX

Profile ID	Model
609	762GX
608	763GX
607	766GX
608	763GX-R
607	766GX-R
607	766GX-R-24
606	767GX
605	711GE
604	716GE

Profile ID	Model
603	717GE
602	721GE
601	726GE
600	727GE
639	725GE
637	742GE
638	743GE
644	744GE

Create AE ONTs

Sample XML create request

The following example shows an XML request for creating an ONT using the FSAN creation method.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="45" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>Ont</type>
              <id>
                <aeontid>0</aeontid>
              </id>
              <aeontid>CXNK23456782</aeontid>
              <reg-id/>
              <subscr-id>Test_7</subscr-id>
              <descr>Customer10Ont</descr>
              <ontprof>
                <type>OntProf</type>
                <id>
                  <ontprof>605</ontprof>
                </id>
              </ontprof>
              <ExternalProvisioned>false</ExternalProvisioned>
              <Region>autodiscovered</Region>
              <BandwidthMetering>vlan</BandwidthMetering>
              <pwe3config>1</pwe3config>
              <timezone>US/Pacific</timezone>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML request for creating an ONT using the RONTA creation method.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="45" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>Ont</type>
              <id>
                <aeontid>0</aeontid>
              </id>
              <aeontid>7775552222</aeontid>
              <reg-id>7775552222</reg-id>
              <subscr-id>Test_7</subscr-id>
              <descr>Customer10Ont</descr>
              <ontprof>
                <type>OntProf</type>
                <id>
                  <ontprof>605</ontprof>
                </id>
              </ontprof>
              <ExternalProvisioned>false</ExternalProvisioned>
              <Region>autodiscovered</Region>
              <BandwidthMetering>vlan</BandwidthMetering>
              <pwe3config>1</pwe3config>
              <timezone>US/Pacific</timezone>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a create request for creating an ONT on an AE ONT. For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`  <aeontid>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
			`<aeontid>` Use 0 to instruct CMS to "auto-generate" the AE ONT number using the value in the <aeontid> tag.

`<aeontid>`	see descr	Yes	`<aeontid>` identifies the ONT using the registration ID or FSAN number of the AE ONT.
`<reg-id>`	Int(10)	see note	Do one of the following: • **FSAN ONT creation method:** Leave this tag empty. When the AE ONT registers, this value (as well as the serial number value) are updated as supplied in the SNMP registration trap. • **RONTA ONT creation method:** Supply the registration ID that is the RONTA identifier. Use the same value as in the <aeontid> tag above.
`<subscr-id>`	Char(63)		AE ONT subscriber ID (not copied to ONT ports).
`<descr>`	Char(48)		AE ONT description (not copied to ONT ports).

`<ontprof>`
`  <type>`
`  <id>`
`  <ontprof>`
	see descr	Yes	`<type>` identifies the profile type as ONT using the following case-sensitive expression: **OntProf**
			`<ontprof>` identifies the ID of a global ONT profile (1 to 50, or one of the default global profiles listed in "Global ONT Profile IDs" on page 279). To view available global profile IDs in CMS Desktop, open the **Profile** > **ONT** > **ONT Profile** Work Area.

Element Tag	Data Type	Req'd ?	Description
<ProvFirmware>	Char		Do one of the following: • Leave empty or do not supply tag to have the system apply the firmware file specified in the Firmware Defaults. To view or set in CMS Desktop, navigate to **System** > **AE Settings**, and click **Action** > **Set Default Firmware**. If no file is specified for the ONT type, the system selects the latest firmware file, if available. • Supply the firmware version to use (for example: calix_ae_700sfu_r2.0.60.1.rto).
<ExternalProvisioned>	Bool	Yes	• **false**—Manage changes to the ONT configuration using CMS. • **true**—Manage changes to ONT configuration using the configuration file on the TFTP server.
<Region>	Char	Yes	Identifies the parent network group in CMS Desktop under which to place the AE ONT (for example: autodiscovered).
<BandwidthMetering>	see descr	Yes	Supply one of the following: • **vlan**—VLAN bandwidth metering • **eth**—Ethernet bandwidth metering
<pwe3config>	see descr	Yes	For business ONTs, supply one of the following: • **1**—Configure PWE3 for T1 services (North America) • **2**—Configure PWE3 for E1 services (International)
<timezone>	see descr	Yes	Supply the time zone for the AE ONT using an IANA standard entry (examples: US/Pacific, America/Vancouver).

XML output element tags

The XML element tags returned from a **create** request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="23" nodename="AeCMSNetwork" username="JDoe"
        sessionid="11">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete" force="true">
              <type>Ont</type>
              <id>
                <aeontid>CXNK00061800</aeontid>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, session ID, and operation), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** request for an AE ONT.

Element Tag	Data Type	Req'd?	Description

`<object operation="delete">`
 	
`<object operation="delete" force="true">`
	Force attribute is Bool	The force attribute is not required.	force="false", or force attribute omitted (default)—Do not perform a force delete. **Note:** For a non-force delete to be successful, all service must be removed from the ONT.
			force="true"—Perform a force delete (deletes that all services on the AE ONT).

`<type>`
`<id>`
`  <aeontid>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
			`<aeontid>` identifies the ONT using the registration ID or FSAN number of the AE ONT.

XML output element tags

The XML element tags returned from a **delete** request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Adding or Updating Subscriber Information on AE ONTs

Subscriber ID and description fields are saved at the AE ONT and AE ONT port (Ethernet, voice, or DS1) levels.

The following examples illustrate how to update the AE ONT subscriber ID and description on a GE and voice port:

- "Update subscriber ID and description on an AE ONT GE port" on page 286
- "Update subscriber ID and description on an AE ONT Voice port" on page 288

In addition, please refer the following sections for more information about how to add or update subscriber information:

- "Creating and Deleting AE ONT ONTs" on page 279
- "Update (merge) subscriber information for gateway service on AE ONTs" on page 312
- "Update (merge) subscriber information for full bridge service on AE ONTs" on page 321

Update subscriber ID and description on an AE ONT GE port

Sample XML subscriber ID request for AE GE port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="72" nodename="AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="30">
   <edit-config>
    <target>
     <running/>
    <target>
    <config>
     <top>
      <object operation="merge">
       <type>OntEthGe</type>
       <id>
        <aeontid>CXNK00078F59</aeontid>
        <ontslot>3</ontslot>
        <ontethge>1</ontethge>
       </id>
       <intf></intf>
       <admin>enabled-no-alarms</admin>
       <subscr-id>daniel-ge-1</subscr-id>
       <descr>daniel-ge-1</descr>
       <sec>
        <type>EthSecProf</type>
        <id>
         <ethsecprof>1</ethsecprof>
        <id>
```

```
      </sec>
      <disable-on-batt>true</disable-on-batt>
      <dscp-prof>
       <type>DscpProf</type>
       <id>
        <dcspprofid>1</dcspprofid>
       <id>
      </dscp-prof>
     </object>
    </top>
   </config>
  </edit-config>
 </rpc>
</soapenv:Body>
</soapenv:Envelope>
```

Sample XML subscriber ID response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply message-id="72" nodename="AeCMSNetwork">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope
```

XML tags and values

The following table lists the XML element tags in a create request for updating subscriber ID and description on an AE ONT GE port.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> </ontslot> <ontethge> </id>	see descr		These element tags identify the port: <type> identifies the provisioning object type using the following case-sensitive expression: **OntEthGe** <aeontid> registration ID or FSAN serial number of the AE ONT. <ontslot>—Enter **3** to identify the Gigabit Ethernet port type <ontethge>—Identifies the GE port number

Element Tag	Data Type	Req'd ?	Description
<admin>	see descr		Operational status of the AE ONT GE port: • **disabled** • **ienabled** • **enabled-no-alarms**
<subscr-id>	Char(63)		AE ONT subscriber ID (not copied to ONT ports).
<descr>	Char(48)		AE ONT description (not copied to ONT ports).
<sec> <type> <id> <ethsecprof> <id> </sec>	see descr		<type> identifies the provisioning object as an Ethernet security profile using the following case-sensitive expression: **EthSecProf** <ethsecprof> identifies the global Ethernet S ecurity profile ID (**1** to **16**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ ONT** > **Security** > **Ethernet**). **Note:** If these tags are not supplied, **1** (system-default) is used for the Ethernet Security profile.

Update subscriber ID and description on an AE ONT Voice port

Sample XML subscriber ID request for AE voice port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="32" nodename="AeCMSNetwork" timeout="35000"
username="rootgod" sessionid="30">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>OntPots</type>
       <id>
        <aeontid>CXNK00078F59</aeontid>
        <ontslot>6</ontslot>
        <ontpots>1</ontpots>
       </id>
       <admin>enabled</admin>
       <subscr-id> daniel-v-1</subscr-id>
       <descr>daniel-v-1</descr>
```

```
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML subscriber ID response for AE voice port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
      <soapenv:Body>
            <rpc-reply message-id="32" nodename="AeCMSNetwork">
                  <ok/>
            </rpc-reply>
      </soapenv:Body>
</soapenv:Envelope>
```

XML tags and values

The following table lists the XML element tags in a create request for updating subscriber ID and description on an AE ONT voice port.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> </id>	see descr		These element tags identify the port: <type> identifies the provisioning object type using the following case-sensitive expression: **OntPots** <aeontid> registration ID or FSAN serial number of the AE ONT. <ontslot>—Enter **3** to identify the Gigabit Ethernet port type <ontpots> identifies the ONT voice port number (1 to 8). **Note:** The number of voice ports varies depending on the ONT model.
<admin>	see descr		Operational status of the AE ONT voice port: • **disabled** • **enabled** • **enabled-no-alarms**
<subscr-id>	Char(63)		AE ONT subscriber ID (not copied to ONT ports).
<descr>	Char(48)		AE ONT description (not copied to ONT ports).

AE ONT Data Service Activation

The XML request defines the objects required to create, update, and delete data services on an AE ONT Ethernet port.

This section contains the following topics:

For read (get-config) requests, see "XML Read (get-config) Requests" on page 271.

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving and Resetting AE ONTs" on page 359.

Create data service on AE ONTs

Sample XML create request

Note: In the following example, XML tags for outer and inner VLANs are not supplied.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="217" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>3</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
```

```
                <bwprof>1</bwprof>
              </id>
            </bw-prof>
            <dhcp-snoop>true</dhcp-snoop>
            <mac-force-forw>true</mac-force-forw>
            <ip-src-verify>true</ip-src-verify>
            <tlan>true</tlan>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating data service on an AE ONT. For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the data service (**1** to **12**; typically **1** to **8** for data service).
<admin>	Char		Operational status of the subscriber port: • **disabled** • **ienabled** • **enabled-no-alarms**
<descr>	Char(31)		Description of service.

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined service tag action (1 to 255).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<dhcp-snoop>	Bool		Use one of the following: • **true**—enable DHCP snooping • **false** (default)—disable DHCP snooping
<mac-force-forw>	Bool		Use one of the following: • **true**—enable MAC forced forwarding • **false** (default)—disable MAC forced forwarding
<ip-src-verify>	Bool		Use one of the following: • **true**—enable IP source verification (binding the IP and MAC addresses to the physical ONT Ethernet port) • **false** (default)—disable IP source verification
<tlan>	Bool		Use one of the following: • **true** (default)—enable multicast filtering of upstream traffic from the ONT Ethernet port. • **false**—disable multicast filtering from the ONT Ethernet port, typically required with Transparent LAN Service (TLS).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) data service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) data service on an AE ONT port is identical to the format for a **create** request (see "XML input element tags and values" on page 291), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>EthSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete data service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="183" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, node name, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** data service request for an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the data service (**1** to **12**; typically **1** to **8** for data service).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="183" nodename="AeCMSNetwork">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** data service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Creating a static IP Address and Subnet Provisioning for an AE ONT Data Service

Note: Updating the static IP through the XML API NBI interface is not supported in CMS.

Sample XML static IP address request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="399" nodename="AeCMSNetwork" username="azhang"
   sessionid="430">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true" >
       <type>EthSvcStaticIp</type>
       <id>
        <aeontid>CXNK0005345C</aeontid>
        <ontslot>3</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <ip-type>ip-addr</ip-type>
       <ip>192.168.1.123</ip>
       <default-gw>192.168.1.1</default-gw>
       <netmask>255.255.255.0</netmask>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="399" nodename="NTWK-E7TRN01">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML element tags

The following table lists the XML element tags required to create a static IP address and subnet request for an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the data service (**1** to **12**; typically **1** to **8** for data service).
<ip-type> <ip> <default-gw> <netmask>	see descr	Yes	<ip-type> configures the IP address: • To create a static IP address, enter the value of <ip-type> element as **ip-addr**. • To create a subnet, enter the value of <ip-type> element as **subnet**. <ip> Enter the static IP address for the ONT port service. <default-gw> Enter the address of the default gateway for subtending static IP address objects. <netmask> Enter the subnet mask for the IP address.

Deleting a static IP Address and Subnet Provisioning for an AE ONT Data Service

Sample request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="1" nodename="NTWK-e7_4_12" username="rootgod"
   sessionid="22">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="delete">
       <type>EthSvcStaticIp</type>
       <id>
        <ont>25</ont>
        <ontslot>3</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
        <ethsvcstaticip>1</ethsvcstaticip>
       </id>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="1" nodename="NTWK-e7_4_12">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

AE ONT Video Service Activation

The XML request defines the objects required to create, update, and delete video services on an AE ONT Ethernet port.

This section contains the following topics:

For read (get-config) requests, see "XML Read (get-config) Requests" on page 271.

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving AE ONT configuration files" on page 359.

Multicast VLAN Registration (MVR) video service

To activate MVR video service, an MVR profile must already be created and configured. The MVR profile is referenced by a multicast profile, which is a required element tag in setting up video service.

Create video service on AE ONTs

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="226" nodename="AeCMSNetwork" username="JDoe"
        sessionid="83">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099332</aeontid>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>9</ethsvc>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>1</svctagaction>
                </id>
              </tag-action>
```

```
            <out-tag>1</out-tag>
            <mcast-prof>
              <type>McastProf</type>
              <id>
                <mcastprof>1</mcastprof>
              </id>
            </mcast-prof>
            <dhcp-snoop>true</dhcp-snoop>
            <mac-force-forw>true</mac-force-forw>
            <ip-src-verify>true</ip-src-verify>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating video service on an AE ONT.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service).
<descr>	Char(31)		Description of service.

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined service tag action (**1** to **255**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (**2** to **4093**, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<admin>	Char		Operational status of the subscriber/video port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** Please refer to the note in "Guidelines for AE ONT Provisioning" on page 266 for more information about the <admin> attributes for AE ONTs.
<mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).
<dhcp-snoop>	Bool		• **true**—enable DHCP snooping • **false** (default)—disable DHCP snooping
<mac-force-forw>	Bool		• **true**—enable MAC forced forwarding • **false** (default)—disable MAC forced forwarding
<ip-src-verify>	Bool		• **true**—enable IP source verification (binding the IP and MAC addresses to the physical ONT Ethernet port) • **false** (default)—disable IP source verification

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="226" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) video service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) video service on an AE ONT port is identical to the format for a **create** request (see "XML input element tags and values" on page 300), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="158" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>EthSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete video service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="198" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099332</aeontid>
                <ontslot>3</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>9</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** video service request on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type using one of the following: • **3**—Gigabit Ethernet port • **4**—HPNA Ethernet port • **5**—Fast Ethernet port <ontethany> identifies the ONT port number (**1** to **8**). <ethsvc> identifies the video service (**1** to **12**; typically **9** to **12** for video service.

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="198" nodename="AeCMSNetwork">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** video service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Applying an ONT Template to an AE ONT

The XML request defines the objects required to apply an ONT template to an AE ONT for residential gateway and full bridge applications on AE ONTs using R3.0 firmware or higher.

Sample request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="424" nodename="AeCMSNetwork"> username=”cmsuser10”
          sessionid=”55”
      <action>
        <action-type>apply-ont-template</action-type>
        <action-args>
          <ontid>
            <type>Ont</type>
            <id>
              <aeontid>CXNK00099335</aeontid>
            </id>
          </ontid>
          <object></object>
          <template-id>
            <type>OntTemplate</type>
            <id>
              <onttmpl>1</onttmpl>
            </id>
          </template-id>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a create request for saving AE ONT configuration files.

Element Tag	Data Type	Req’d ?	Description
<action> <action-type> <action-args>	see descr	Yes	These element tags identify the request type, as shown in the example request. For the <action-type> tag, use the following case-sensitive expression: **apply-ont-template**

Element Tag	Data Type	Req'd ?	Description
<ontid> <type> <id> <aeontid>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <aeontid> identifies the AE ONT configuration files to be saved using the registration ID or FSAN number.
<template-id> <type> <id> <onttmpl>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntTemplate** <onttmpl> identifies a global ONT template ID (**1** to **20**).

AE ONT Gateway Service Activation

The XML request defines the objects required to create, update, and delete residential gateway (RG) services on an AE ONT Ethernet port. Additionally, subscriber ID and description tags can be updated using a unique XML request.

Note: Provisioning AE ONT gateway service requires AE ONT firmware 3.0 or higher.

Before provisioning service, you can optionally apply an ONT template to assign settings to the ONT's GE ports. See "Applying an ONT Template to an AE ONT" on page 305.

This section contains the following topics:

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving AE ONT configuration files" on page 359.

Create gateway service on AE ONTs

Sample XML create request

Note: In the following example, XML tags for outer and inner VLANs are not supplied.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="203" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>8</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>3</svctagaction>
                </id>
```

```
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof>1</bwprof>
              </id>
            </bw-prof>
            <mcast-prof>
              <type>McastProf</type>
              <id>
                <mcastprof>1</mcastprof>
              </id>
            </mcast-prof>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating gateway service on an AE ONT. For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **8**—Gateway <ontethany> identifies the ONT RG ID (**1**). <ethsvc> identifies the service (**1** to **12**).

Element Tag	Data Type	Req'd ?	Description
<admin>	Char		Operational status of the subscriber/video port: • **disabled** • **ienabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled-no-alarms) is used. **Note:** Please refer to the note in "Guidelines for AE ONT Provisioning" on page 266 for more information about the <admin> attributes for AE ONTs.
<descr>	Char(31)		Description of service.
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined service tag action (**1** to **255**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (**2** to **4093**, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
```

```
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) gateway service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) residential gateway service on an AE ONT port is identical to the format for a **create** request (see "XML input element tags and values" on page 308), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>8</ontslot>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) subscriber information for gateway service on AE ONTs

Sample XML update request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="AeCMSNetwork" username="JDoe"
        sessionid="288">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntRg</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>8</ontslot>
                <ongrg>1</ontrg>
              </id>
              <subscr-id>707-766-3500</subscr-id>
              <descr>Bedrock Apts.</descr>
              <admin>enabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

The following table lists the XML element tags in a **update** request for updating the subscriber information for gateway service on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontrg>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntRg** <aeontid> identifies the ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **8**—Gateway <ontrg> identifies the RG object (**1**).
<subscr-id>	Char(31)		Subscriber ID.
<descr>	Char(31)		Description of service.
<admin>	Char		Operational status of the subscriber port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.

Delete gateway service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id=”183” nodename=”AeCMSNetwork” username=”cmsuser10”
        sessionid=”55”>
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation=”delete”>
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>8</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, node name, user name, and session ID), see “Common XML Element Tags for Edit-Config Requests” on page 267.

The following table lists the XML element tags required in a **delete** gateway service request for an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **8**—Gateway service <ontethany> identifies the ONT RG ID (**1**). <ethsvc> identifies the gateway service (**1** to **12**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="183" nodename="AeCMSNetwork">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** gateway service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

AE ONT Full Bridge Service Activation

The XML request defines the objects required to create, update, and delete full bridge services on an AE ONT Ethernet port. Additionally, subscriber ID and description tags can be updated using a unique XML request.

Note: Provisioning AE ONT full bridge service requires AE ONT firmware 3.0 or higher.

Before provisioning service, you can optionally apply an ONT template to assign settings to the ONT's GE ports. See "Applying an ONT Template to an AE ONT" on page 305.

This section contains the following topics:

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving AE ONT configuration files" on page 359.

Create full bridge services on AE ONTs

Sample XML create request

In the following example, a full bridge data service request is sent with an XML tag for the outer VLAN.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="245" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>9</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>3</svctagaction>
                </id>
```

```
            </tag-action>
            <bw-prof>
              <type>BwProf</type>
              <id>
                <bwprof>1</bwprof>
              </id>
            </bw-prof>
            <out-tag>123</out-tag>
          </object>
        </top>
      </config>
    </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

In the following example, a full bridge video service request is sent with an XML tag for the outer VLAN.

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="245" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="create">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>9</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>9</ethsvc>
              </id>
              <admin>enabled</admin>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>3</svctagaction>
                </id>
              </tag-action>
              <bw-prof>
                <type>BwProf</type>
                <id>
                  <bwprof>1</bwprof>
                </id>
              </bw-prof>
              <mcast-prof>
                <type>McastProf</type>
                <id>
                  <mcastprof>1</mcastprof>
                </id>
              </mcast-prof>
              <out-tag>123</out-tag>
```

```
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating full bridge service on an AE ONT. For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **9**—Full Bridge <ontethany> identifies the ONT FB ID (**1**). <ethsvc> identifies the service (**1** to **12**)
<admin>	Char		Operational status of the subscriber port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.
<descr>	Char(31)		Description of service.
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined service tag action (**1** to **255**).

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<bw-prof> <type> <id> <bwprof>	see descr	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
Additional tags required for video service: <mcast-prof> <type> <id> <mcastprof>	see descr	Yes	<type> identifies the profile type as multicast using the following case-sensitive expression: **McastProf** <mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).
<dhcp-snoop>	Bool		Use one of the following: • **true**—enable DHCP snooping • **false**—disable DHCP snooping (default)
<mac-force-forw>	Bool		Use one of the following: • **true**—enable MAC forced forwarding • **false**—disable MAC forced forwarding (default)
<ip-src-verify>	Bool		Use one of the following: • **true**—enable IP source verification (binding the IP and MAC addresses to the physical ONT Ethernet port) • **false**—disable IP source verification (default)

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="217" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) full bridge service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) residential service on an AE ONT port is identical to the format for a **create** request (see "XML input element tags and values" on page 308), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>9</ontslot>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) subscriber information for full bridge service on AE ONTs

Sample XML update request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="148" nodename="AeCMSNetwork" username="JDoe"
        sessionid="288">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntFb</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>9</ontslot>
                <ongfb>1</ontfb>
              </id>
              <subscr-id>707-766-3500</subscr-id>
              <descr>Bedrock Apts.</descr>
              <admin>enabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

The following table lists the XML element tags in a **update** request for updating the subscriber information for full bridge service on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontfb>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **OntFb** <aeontid> identifies the ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **9**—Full Bridge <ontrg> identifies the FB object (**1**).
<subscr-id>	Char(31)		Subscriber ID.
<descr>	Char(31)		Description of service.
<admin>	Char		Operational status of the subscriber port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.

Delete full bridge service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id=”183” nodename=”AeCMSNetwork” username=”cmsuser10”
        sessionid=”55”>
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation=”delete”>
              <type>EthSvc</type>
              <id>
                <aeontid>CXNK00099331</aeontid>
                <ontslot>9</ontslot>
                <ontethany>1</ontethany>
                <ethsvc>1</ethsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, node name, user name, and session ID), see “Common XML Element Tags for Edit-Config Requests” on page 267.

The following table lists the XML element tags required in a **delete** full bridge service request for an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **EthSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type: • **9**—Full Bridge <ontethany> identifies the ONT FB ID (**1**). <ethsvc> identifies the service (**1** to **12**).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="183" nodename="AeCMSNetwork">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** full bridge service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

AE ONT Voice Service Activation

The XML request defines the objects required to create, update, and delete SIP, TDM gateway, and H.248 VoIP services on AE ONT voice ports.

Before voice service can be activated, you must update the IP host for each type of voice service (SIP, TDM gateway, or H.248) for each AE ONT.

This section contains the following topics:

For read (get-config) requests, see "XML Read (get-config) Requests" on page 271.

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving AE ONT configuration files" on page 359.

Update (merge) an IP host for VoIP service

Sample XML merge requests

The following example is for an IP host for SIP service using DHCP host protocol.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="333" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntIpHost</type>
              <id>
                <aeontid>CXNK00099338</aeontid>
                <ontiphost>1</ontiphost>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>18</svctagaction>
                </id>
              </tag-action>
              <host-proto>dhcp</host-proto>
              <gwf>true</gwf>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example is for an IP host for TDM gateway service with static host protocol.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="334" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>OntIpHost</type>
              <id>
                <aeontid>CXNK00099337</aeontid>
                <ontiphost>2</ontiphost>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
```

```
                <id>
                  <svctagaction>5</svctagaction>
                </id>
              </tag-action>
              <host-proto>static</host-proto>
              <static-ip>192.11.14.101</static-ip>
              <static-mask>255.255.255.0</static-mask>
              <static-gw>192.11.14.102</static-gw>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for configuring a static host for AE ONT voice service.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`  <aeontid>`
`  <ontiphost>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **OntIpHost**
			`<aeontid>` identifies the AE ONT by its registration number or FSAN serial number.
			`<ontiphost>` identifies the VoIP service type using one of the following:
			• **1**—SIP service
			• **2**—TDM gateway service
			• **4**—H.248 service
			• **5**—MGCP service

`<tag-action>`
`  <type>`
`  <id>`
`  <svctagaction>`
	see descr	Yes	`<type>` identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction**
			`<svctagaction>` identifies the ID of a pre-defined service tag action

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<host-proto>	see descr	Yes	Use one of the following: • **dhcp**—DHCP host protocol • **static**—Static host protocol • **unknown**
<gwf>	Bool		• **true** (default for SIP and TDM Gateway)—Force the use of the AE ONT management VLAN for voice service. • **false**—(default for H.248 and MGCP Gateway) Use a voice service VLAN.
Additional element tag for H.248 and MGCP with DHCP host protocol: <hostname>	Char	Yes	Fully qualified domain name. of the DHCP host server, required only for H.248 and MGCP service when <hostprotocol> is **dhcp**.
Additional element tags for static host protocol: <static-ip> <static-mask> <static-gw>	see descr	Yes	Use the following for these three additional element tags: • static IP address assigned to the ONT, in dotted quad format. • static IP mask assigned to the ONT, in dotted quad format. • static IP gateway for the ONT to use in routing traffic, in dotted quad format (must belong to the same subnet as the subtending IP address) **Note:** When the host protocol is DHCP, these element tags are not used (though the values are saved in the service record).

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Create SIP VoIP service on AE ONTs

Before SIP voice service can be activated for the first time, you must update the IP host on the AE ONT. See "Update (merge) an IP host for VoIP service" on page 326.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="240" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>SipSvc</type>
              <id>
                <aeontid>CXNK00099341</aeontid>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <sipsvc>1</sipsvc>
              </id>
              <sip-prof>
                <type>SipProf</type>
                <id>
                  <sipprof>1</sipprof>
                </id>
              </sip-prof>
              <admin>enabled</admin>
              <user>West_sip_svc</user>
              <passwd>test</passwd>
              <uri>uri</uri>
              <call-waiting>false</call-waiting>
              <caller-id-enabled>true</caller-id-enabled>
              <three-way-calling>true</three-way-calling>
              <t38-fax-relay>false</t38-fax-relay>
              <sip-rmt-cfg-override>false</sip-rmt-cfg-override>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating SIP VoIP service on an AE ONT.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **SipSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). **Note:** The number of voice ports varies depending on the ONT model. <sipsvc> identifies the voice service (1).
<sip-prof> <type> <id> <sipprof>	see descr	Yes	<type> identifies the profile type using the following case-sensitive expression: **SipProf** <sipprof> identifies the ID of a pre-defined global SIP Gateway profile (**1** to **512**).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.
<user>	Char	Yes	User name.
<passwd>	Char	Yes	Password for specified user.
<uri>	Char	Yes	Universal resource identifier for SIP service.

Element Tag	Data Type	Req'd ?	Description
<call-waiting> <caller-id-enabled> <three-way-calling> <t38-fax-relay>	Bool.		These parameters apply to AE ONTs operating with R3.0 or higher firmware. Valid values are: • **true—**Enabled • **false**—Disabled (default) <call-waiting> Call Waiting feature <caller-id-enabled> Caller ID feature <three-way-calling> Three-Way Calling feature <t38-fax-relay> T38 Fax Relay feature
<sip-rmt-cfg-override>	Bool.		Override flag for SIP remote service configuration. Applies to AE ONTs operating with R3.0 or higher firmware: • **true—**use the SIP configuration file • **false**—use the local configuration (default)

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="240" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) SIP voice service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) SIP voice service on an AE port is identical to the format for a **create** request (see "XML input element tags and values" on page 330), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="230" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>SipSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete SIP VoIP service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="806" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>SipSvc</type>
              <id>
                <aeontid>CXNK00099341</aeontid>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <sipsvc>1</sipsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** SIP VoIP service request on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <sipsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **SipSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). <sipsvc> identifies the voice service (1).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="806" nodename="AeCMSNetwork">
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** SIP VoIP service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Create TDM Gateway VoIP service on AE ONTs

Before TDM gateway voice service can be activated for the first time, you must update the IP host on the AE ONT. See "Update (merge) an IP host for VoIP service" on page 326.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1023" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>TdmGwSvc</type>
              <id>
                <aeontid>CXNK00099351</aeontid>
                <ontslot>6</ontslot>
                <ontpots>3</ontpots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
              <tdmgw-prof>
                <type>TdmGwProf</type>
                <id>
                  <tdmgwprof>1</tdmgwprof>
                </id>
              </tdmgw-prof>
              <admin>enabled</admin>
              <crv>N1-1-IG1-224</crv>
              <tdmgw-svc-group>
                <type>TDMGWServiceGroup</type>
                <id>
                  <tdmgwservicegroup>8</tdmgwservicegroup>
                </id>
              </tdmgw-svc-group>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating TDM gateway VoIP service on an AE ONT.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **TdmGwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). **Note:** The number of ONT voice ports varies based on the ONT model. <tdmgwsvc> identifies the voice service (1).
<tdmgw-prof> <type> <id> <tdmgwprof>	see descr	Yes	<type> identifies the profile type as TDM gateway using the following case-sensitive expression: **TdmGwProf** <tdmgwprof> identifies the ID of a pre-defined global TDM gateway profile (1 to 32).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.

Element Tag	Data Type	Req'd ?	Description
<crv>	Char	Yes	Call Reference Value for the subscriber line in GR-303 or GR-8 switch interface group. Example: **N1-1-IG1-224** Note the following: • The CRV must be provisioned on the C7 gateway interface group. • The format is case-sensitive, and must be upper case.
<tdmgw-svc-group> <type> <id> <tdmgwservicegroup>	see descr		<type> identifies the object type using the following case-sensitive expression: **TDMGWServiceGroup** <tdmgwservicegroup> identifies the ID of a pre-defined CMS TDM gateway service group (**1** to **1000**). **Note:** If elements tags for the TDM service group are not supplied, the XML reply does not include them.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1023" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) TDM gateway voice service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) TDM gateway voice service on an AE port is identical to the format for a **create** request (see "XML input element tags and values" on page 336), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="231" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>TdmGwSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete TDM Gateway VoIP service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="810" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target><running/></target>
        <config>
          <top>
            <object operation="delete">
              <type>TdmGwSvc</type>
              <id>
                <aeontid>CXNK00099351</aeontid>
                <ontslot>6</ontslot>
                <ontpots>2</ontpots>
                <tdmgwsvc>1</tdmgwsvc>
              </id>
            </object>
```

```
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** TDM gateway VoIP service request on an AE ONT.

Element Tag	Data Type	Req'd?	Description
<type> <id> <aeontid> <ontslot> <ontds1> <tdmgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **TdmGwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). <tdmgwsvc> identifies the service type (1).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="810" nodename="AeCMSNetwork">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** TDM gateway VoIP service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269

Create H.248 VoIP service on AE ONTs

Before H.248 voice service can be activated for the first time, you must update the IP host on the AE ONT. See "Update (merge) an IP host for VoIP service" on page 326.

Sample XML create request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1130" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>H248GwSvc</type>
              <id>
                <aeontid>CXNK00099371</aeontid>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <h248gwsvc>1</h248gwsvc>
              </id>
              <h248gw-prof>
                <type>H248GwProf</type>
                <id>
                  <h248gwprof>1</h248gwprof>
                </id>
              </h248gw-prof>
              <termination-id>1</termination-id>
              <admin>enabled</admin>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating H.248 VoIP service on an AE ONT.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <h248gwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **H248GwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (**6**). <ontpots> identifies the ONT voice port number (**1** to **8**). N**ote:** The number of voice ports varies depending on the ONT model. <h248gwsvc> identifies the service type (**1**).
<h248gw-prof> <type> <id> <h248gwprof>	see descr	Yes	<type> identifies the profile as an H.248 type using the following case-sensitive expression: **H248GwProf** <h248gwprof> identifies the ID of a pre-defined global H.248 profile (**1** to **20**).
<termination-id>	Char	Yes	Identifies the H.248 termination ID.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1130" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) H.248 voice service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) H.248 VoIP service on an AE port is identical to the format for a **create** request (see "XML input element tags and values" on page 340), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="232" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>H248GwSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete H.248 VoIP service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="811" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>H248GwSvc</type>
              <id>
                <aeontid>CXNK00099371</aeontid>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <h248gwsvc>1</h248gwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** H.248 VoIP service request on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <h248gwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **H248GwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). <h248gwsvc> identifies the service type (1 to 20).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="811" nodename="AeCMSNetwork">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** H.248 VoIP service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269

Create MGCP VoIP service on AE ONTs

Before MGCP voice service can be activated for the first time, you must update the IP host on the AE ONT. See "Update (merge) an IP host for VoIP service" on page 326.

Sample XML create request

```
soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1244" nodename="AeCMSNetwork" username="rootgod"
        sessionid="1">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>MgcpGwSvc</type>
              <id>
                <aeontid>CXNK00000725</aeontid>
                <ontslot>6</ontslot>
                <ontpots>1</ontpots>
                <mgcpgwsvc>1</mgcpgwsvc>
              </id>
              <mgcpgw-prof>
                <type>MgcpGwProf</type>
                <id>
                  <mgcpgwprof>2</mgcpgwprof>
                </id>
              </mgcpgw-prof>
              <termination-id>1</termination-id>
              <gr-303>false</gr-303>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating MGCP VoIP service on an AE ONT.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <mgcpgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **MgcpGwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). **Note:** The number of voice ports varies depending on the ONT model. <mgcpgwsvc> identifies the service type (1).
<mgcpgw-prof> <type> <id> <mgcpgwprof>	see descr	Yes	<type> identifies the profile as an MGCP type using the following case-sensitive expression: **MgcpGwProf** <mgcpgwprof> identifies the ID of a pre-defined global MGCP profile (1 to 20).
<termination-id>	Char	Yes	Identifies the MGCP termination ID.
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.
<gr-303>	Bool		(Requires AE ONT firmware R3.0 or higher.) Enables or disables RFC 2833 voice control protocol compatible with the B6-640 Trunking Gateway (TGW) in GR-303 mode. Valid values: • **true—**Enabled • **false**—Disabled (default)

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1224" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) MGCP voice service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) MGCP VoIP service on an AE port is identical to the format for a **create** request (see "XML input element tags and values" on page 346), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="235" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>MgcpGwSvc</type>
                ...
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete MGCP VoIP service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="814" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="65">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>MgcpGwSvc</type>
              <id>
                <aeontid>CXNK00099371</aeontid>
                <ontslot>6</ontslot>
                <ontpots>4</ontpots>
                <mgcpgwsvc>1</mgcpgwsvc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** MGCP VoIP service request on an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontpots> <mgcpgwsvc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **MgcpGwSvc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (6). <ontpots> identifies the ONT voice port number (1 to 8). <mgcpgwsvc> identifies the service type (1 to 20).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="811" nodename="AeCMSNetwork">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** MGCP VoIP service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269

AE ONT PWE3 Service Activation

The XML request defines the objects required to create, update, and delete PWE3 services on an AE ONT T1 port.

Before voice service can be activated, you must update the IP host for PWE3 service for each AE ONT.

This section contains the following topics:

For read (get-config) requests, see "XML Read (get-config) Requests" on page 271.

After provisioning services, the AE ONT configuration files must be saved and the AE ONT must be reset. For instructions, see "Saving AE ONT configuration files" on page 359.

Update (merge) an IP host for PWE3 DS1 service

Sample XML merge request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="334" nodename="AeCMSNetwork" username="JDoe"
        sessionid="65">
      <edit-config>
        <target><running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>OntIpHost</type>
              <id>
                <aeontid>CXNK00099391</aeontid>
                <ontiphost>4</ontiphost>
              </id>
              <tag-action>
                <type>SvcTagAction</type>
                <id>
                  <svctagaction>5</svctagaction>
                </id>
              </tag-action>
              <out-tag>none</out-tag>
              <in-tag>none</in-tag>
              <host-proto>static</host-proto>
              <static-ip>192.11.14.101</static-ip>
              <static-mask>255.255.255.0</static-mask>
              <static-gw>192.11.14.102</static-gw>
```

```
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for configuring a static host for AE ONT voice service.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`  <aeontid>`
`  <ontiphost>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **OntIpHost**
			`<aeontid>` identifies the AE ONT by its registration number or FSAN serial number.
			`<ontiphost>` identifies the service type using the following:
			• **3**—PWE3 service

`<tag-action>`
`  <type>`
`  <id>`
`  <svctagaction>`
	see descr	Yes	`<type>` identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction**
			`<svctagaction>` identifies an ID of a pre-defined service tag action (1 to 255).

`<out-tag>` `<in-tag>`	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the service tag action, do not include the element tag in the XML request.
`<hostprotocol>`	see descr	Yes	Use the following: • **static**—Static host protocol

Element Tag	Data Type	Req'd ?	Description
<static-ip> <static-mask> <static-gw>	see descr	Yes	Use the following for these three additional element tags: • static IP address assigned to the ONT, in dotted quad format. • static IP mask assigned to the ONT, in dotted quad format. • static IP gateway for the ONT to use in routing traffic, in dotted quad format (must belong to the same subnet as the subtending IP address) **Note:** When the host protocol is DHCP, these element tags are not used (though the values are preserved in the service record).

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Create PWE3 service on AE ONTs

Sample XML create request

The following example shows an XML request with UDP-IP transport mode.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="65" nodename="AeCMSNetwork" username="rootgod"
        sessionid="1">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="create">
              <type>Pwe3Svc</type>
              <id>
                <aeontid>CXNK00099335</aeontid>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
              <ds1-pwe3-prof>
                <type>Ds1Pwe3Prof</type>
                <id>
                  <ds1pwe3prof>3</ds1pwe3prof>
                </id>
              </ds1-pwe3-prof>
              <admin>enabled</admin>
              <udp-port>30001</udp-port>
              <remote-ip>2.2.9.7</remote-ip>
              <transport>udp-ip</transport>
              <remote-mac>00:00:00:00:00:00</remote-mac>
              <remote-udp>30001</remote-udp>
              <out-tag>4095</out-tag>
              <in-tag>4095</in-tag>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML request with MEF transport mode.

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="65" nodename="AeCMSNetwork" username="rootgod"
        sessionid="1">
      <edit-config>
        <target>
          <running/>
```

```
        </target>
        <config>
          <top>
            <object operation="create">
              <type>Pwe3Svc</type>
              <id>
                <aeontid>CXNK00099335</aeontid>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
              <ds1-pwe3-prof>
                <type>Ds1Pwe3Prof</type>
                <id>
                  <ds1pwe3prof>3</ds1pwe3prof>
                </id>
              </ds1-pwe3-prof>
              <admin>enabled</admin>
              <transport>mef</transport>
              <remote-ip>191.23.43.33</remote-ip>
              <ecid>3300</ecid>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating PWE3 service on an AE ONT. For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontds1> <pwe3svc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Pwe3Svc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (7). <ontds1> identifies the ONT T1/E1 port number (1 to 8). N**ote:** The number of T1/E1 ports varies depending on the ONT model. <pwe3svc> identifies the service type (1).

Element Tag	Data Type	Req'd ?	Description
<ds1-pwe3-prof> <type> <id> <ds1pwe3prof>	see descr	Yes	<type> identifies the profile type as DS1 PWE3 using the following case-sensitive expression: **DS1Pwe3Prof** <ds1pwe3prof> identifies the ID of a pre-defined global T1/E1 PWE3 profile (**1** to **8**).
<admin>	Char		Operational status of the subscriber/voice port: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the default operational status (enabled) is used.
<transport>	Char		Identifies the transport mode using one of the following: • **udp-ip** (default, if element tag is not supplied) • **mef**
<udp-port>	Int	in UPD-IP mode	(Required in UDP-IP transport mode) Local UDP port number (1024 to 65534).
<remote-udp>	Int		(Optional in UDP-IP transport mode) Remote UDP port number (1024 to 65534).
<remote-IP>	IP address format	see note	IP address of the remote end of the pseudowire, in dotted quad format. Alternately, supply **none** to reset the value to 0.0.0.0. **Note:** In UDP-IP transport mode, this element tag is required. In MEF transport mode, either this element tag or the remote-mac element tag is required.
<remote-mac>	MAC format	see note	(Applicable for MEF transport mode) The MAC address of the remote end of the pseudowire. The data format is six hexadecimal digits in the range 0-FF, optionally separated by colons. **Note:** In MEF transport mode, either this element tag or the remote-ip element tag must be supplied.
<ecid>	Int		(Optional in MEF transport mode) Emulated Circuit ID (0 to 1048575).
<remote-ecid>	Int		(Optional in MEF transport mode) Remote Emulated Circuit ID (0 to 1048575).

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs), when required for the service. **Note:** When an outer or inner tag is not required, use 4095, or do not include the element tag in the request.

Sample XML create reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1414" nodename="AeCMSNetwork"
      <ok/>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Update (merge) PWE3 service on AE ONTs

XML merge requests, input element tags, and output element tags

The format for the XML request to update (merge) PWE3 service on an AE port is identical to the format for a **create** request (see "XML input element tags and values" on page 354), with one exception: For the operation attribute in the object tag, use **merge** instead of **create**, as shown in the following XML request excerpt:

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1415" nodename="AeCMSNetwork" username="JDoe"
        sessionid="37">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="merge">
              <type>Pwe3Svc</type>
                . . .
```

Important Note: Even if a service parameter value is not changing, the corresponding XML element tag must be supplied in the request.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267 and "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Delete PWE3 service on AE ONTs

Sample XML delete request

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="1416" nodename="AeCMSNetwork" username="cmsuser10"
        sessionid="55">
      <edit-config>
        <target>
          <running/>
        </target>
        <config>
          <top>
            <object operation="delete">
              <type>Pwe3Svc</type>
              <id>
                <aeontid>CXNK00099335</aeontid>
                <ontslot>7</ontslot>
                <ontds1>1</ontds1>
                <pwe3svc>1</pwe3svc>
              </id>
            </object>
          </top>
        </config>
      </edit-config>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

For descriptions of the common XML element tags supplied in the request (message ID, nodename, user name, and session ID), see "Common XML Element Tags for Edit-Config Requests" on page 267.

The following table lists the XML element tags required in a **delete** PWE3 service request for an AE ONT.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontds1> <pwe3svc>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Pwe3Svc** <aeontid> identifies the AE ONT by its registration number or FSAN serial number. <ontslot> identifies the ONT port type (7). <ontds1> identifies the ONT T1/E1 port number (1 to 8). N**ote:** The number of T1/E1 ports varies depending on the ONT model. <pwe3svc> identifies the service type (1).

Sample XML delete reply

```
<soapenv:Envelope xmlns:soapenv="www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc-reply message-id="1416" nodename="AeCMSNetwork">
      <ok/>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The XML element tags returned from a **delete** PWE3 service request for an AE ONT include the message ID and node name, along with the status of the request.

For descriptions of common XML element tags, see "Common element tags for AE ONT edit-configuration XML replies" on page 269.

Saving and Resetting AE ONTs

The XML request defines the objects required to save and reset AE ONTs.

This section contains the following topics:

Saving AE ONT configuration files

Sample XML save AE ONT request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="246" nodename="AeCMSNetwork"> username=”cmsuser10”
          sessionid=”55”
      <action>
        <action-type>commit-aeont</action-type>
        <action-args>
          <ontid>
            <type>Ont</type>
            <id>
              <aeontid>CXNK00099335</aeontid>
            </id>
          </ontid>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a create request for saving AE ONT configuration files.

Element Tag	Data Type	Req'd ?	Description
<action> <action-type> <action-args>	see descr	Yes	These element tags identify the request type, as shown in the example request. For the <action-type> tag, use the following case-sensitive expression: **commit-aeont**

Element Tag	Data Type	Req'd ?	Description
<ontid> <type> <id> <aeontid>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <aeontid> identifies the AE ONT configuration files to be saved using the registration ID or FSAN number.

Resetting AE ONTs

Sample XML reset request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Body>
    <rpc message-id="243" nodename="AeCMSNetwork"> username=”cmsuser10”
          sessionid=”55”
      <action>
        <action-type>reset-aeont</action-type>
        <action-args>
          <ontid>
            <type>Ont</type>
            <id>
              <aeontid>CXNK00099335</aeontid>
            </id>
          </ontid>
        </action-args>
      </action>
    </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following table lists the XML element tags in a create request for resetting an AE.

Element Tag	Data Type	Req'd ?	Description
<action> <action-type> <action-args>	see descr	Yes	These element tags identify the request type, as shown in the example request. For the <action-type> tag, use the following case-sensitive expression: **reset-aeont**
<ontid> <type> <id> <aeontid>	see descr	Yes	<type> identifies the provisioning object type using the following case-sensitive expression: **Ont** <aeontid> identifies the AE ONT to be reset using the registration ID or FSAN number.

Configuring Power Shedding on an ONT GE Port

Power shedding is controlled at the ONT port level and can be set to determine whether services (ports) remain on or are disabled when an ONT goes on battery back-up power. This is done by configuring the **disable-on-batt**" value as true or false for the GE port.

- When the **disable-on-batt** field is set to **true**, the GE port is disabled after five minutes, allowing critical applications to be shut down gracefully.
- When the **disable-on-batt** field is set to **false**, Ethernet port services remain active indefinitely, or until the battery capacity is depleted.

Obtaining AE ONT Port Values

Sample XML request for port provisioning values

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="113" nodename="AeCMSNetwork" timeout="35000">
   <get-config>
    <source>
     <running/>
    <source>
    <filter type="subtree">
     <top>
      <object>
       <type>OntEthGe</type>
       <id>
         <ont>7</ont>
         <ontslot>3</ontslot>
         <ontethge>1</ontethge>
        </id>
       </object>
      </top>
     </filter>
   </get-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response for port provisioning values

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply message-id="113" nodename="AeCMSNetwork">
   <data>
    <top>
      <object>
        <type>OntEthGe</type>
        <id>
         <ont name="CXNK0004B7F1">7</ont>
         <ontslot>3</ontslot>
```

```
          <ontethge>1</ontethge>
         </id>
         <intf></intf>
         <admin>ienabled</admin>
         <subscr-id>E7TRN01_AEMDUONT_2G7</subscr-id>
         <descr>E7TRN01_AEMDUONT_2G7</descr>
         <gos></gos>
         <duplex>full</duplex>
         <sec>
          <type>EthSecProf</type>
          <id>
           <ethsecprof name="system-default">1</ethsecprof>
          </id>
         </sec>
         <disable-on-batt>true</disable-on-batt>
         <dscp-prof>
          <type>DscpProf</type>
          <id>
           <dcspprofid name="system-default">1</dcspprofid>
          </id>
         </dscp-prof>
         <speed>auto</speed>
        </object>
       </top>
      </data>
     </rpc-reply>
    </soapenv:Body>
</soapenv:Envelope>
```

Enabling or disabling the battery option

Sample XML request to disable the battery option

In this example the <*disable-on-batt*> field is set to "**true**", disabling the FE/GE port five minutes after the ONT goes on battery back-up power

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="21" nodename="AeCMSNetwork" timeout="35000"
username=”rootgod” sessionid=”392”>
   <edit-config>
    <target>
     <running/>
    <target>
     <config>
      <top>
       <object operation="merge">
         <type>OntEthGe</type>
         <id>
          <aeontid>CXNK12345678</aeontid>
          <ontslot>3</ontslot>
          <ontethge>1</ontethge>
         </id>
```

```
          <admin>ienabled</admin>
          <subscr-id>E7TRN01_AEMDUONT_2G7</subscr-id>
          <descr>E7TRN01_AEMDUONT_2G7</descr>
          <duplex>full</duplex>
          <sec>
           <type>EthSecProf</type>
            <id>
             <ethsecprof>1</ethsecprof>
            </id>
           </sec>
           <disable-on-batt>true</disable-on-batt>
           <dscp-prof>
            <type>DscpProf</type>
             <id>
              <dcspprofid>1</dcspprofid>
             </id>
            </dscp-prof>
            <speed>auto</speed>
       </object>
      </top>
     </config>
    <edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML tags and values

The following table lists the XML element tags in a create request for enabling or disabling the battery option.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <aeontid> <ontslot> <ontetge>	see descr	Yes	These element tags identify the port: <type> identifies the provisioning object type using the following case-sensitive expression: **OntEthGe** <aeontid> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—Enter **3** to identify the Gigabit Ethernet port type <ontethge>—Identifies the GE port number

Element Tag	Data Type	Req'd ?	Description
<admin>	see descr	No	Operational status of the AE ONT GE port: • **disabled** • **ienabled** (The value "ienabled" is used for the ONT Ethernet port admin state) • **enabled-no-alarms)** **Note:** If this element tag is not supplied in the XML request, **enabled-no-alarms** is used. **Note:** Please refer to "Guidelines for AE ONT Provisioning" on page 266 for more information.
<subscr-id>	Char(63)	No	AE ONT subscriber ID (not copied to ONT ports).
<descr>	Char(48)	No	AE ONT description (not copied to ONT ports).
<duplex>	Bool	Yes	Identifies the duplex mode for the Ethernet port: • **full** (default if no tag is provided) **half**
<sec> <type> <id> <ethsecprof>	see descr	Yes	<type> identifies the provisioning object as an Ethernet security profile using the following case-sensitive expression: **EthSecProf** <ethsecprof> identifies the global Ethernet Security profile ID (**1** to **16**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **Security** > **Ethernet**). **Note:** If these tags are not supplied, **1** (system-default) is used for the Ethernet Security profile.

Element Tag	Data Type	Req'd ?	Description
</sec> <disable-on-batt>	see descr	Yes	<disable-on-batt> - determine whether services (ports) remain on or are disabled when an ONT goes on battery back-up power. • **true**: The GE port is disabled after five minutes, allowing critical applications to be shut down gracefully. This is the default value. • **false**: Ethernet port services remain active indefinitely, or until the battery capacity is depleted.
<dscp-prof> <type> <id> <dcsppprofid>	see descr	Yes	Specify an DSCP Map profile. **Note:** The <dscp-prof> element is used for an AEONT. The <dscp-map> element is used for an E7. <dscp-prof> <type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpProf** <dscpprofiid> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.
<speed>	see descr	Yes	Identifies the data rate of the Ethernet port: • **auto**—auto negotiate the rate (default if no tag is provided) • **1000—1000 Mbps**

AE ONT Dynamic Load Action

AE Release 3.1 supports Dynamic AE Management, which allows ONTs to maintain services while changes to the provisioning state are being applied.

Note: To use the AE R3.1 Dynamic Provisioning feature, the "Enable Dynamic Provisioning" flag must be enabled in the configuration file. The feature may not work properly if this flag is not enabled.

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="6" nodename="NTWK-AeCMSNetwork" timeout="35000"
   username="rootgod" sessionid="122">
   <action>
    <action-type>dynamic-load</action-type>
    <action-args>
     <ontid>
      <type>Ont</type>
      <id>
       <aeontid>CXNK000B3A6C</aeontid>
      </id>
     </ontid>
    </action-args>
   </action>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc-reply message-id="6" nodename="AeCMSNetwork">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

Provisioning 844GE Default WAN Service

Use the following commands to set the RG default WAN service on an 844GE.

Sample XML Request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
     <soapenv:Body>
        <rpc message-id="43" nodename="AeCMSNetwork" username="rootgod"
sessionid="13">
           <edit-config>
              <target>
                 <running/>
              </target>
              <config>
                 <top>
                    <object operation="merge">
                       <type>OntRg</type>
                       <id>
                          <ont>2</ont>
                          <ontslot>8</ontslot>
                          <ontrg>1</ontrg>
                       </id>
                       <admin>enabled-no-alarms</admin>
                       <subscr-id/>
                       <descr/>
                       <mgmt-mode>native</mgmt-mode>
                       <wan-protocol>dhcp</wan-protocol>
                       <static-ip>0.0.0.0</static-ip>
                       <static-ip-mask>0.0.0.0</static-ip-mask>
                       <static-ip-gw>0.0.0.0</static-ip-gw>
                       <pri-dns-server>0.0.0.0</pri-dns-server>
                       <sec-dns-server>0.0.0.0</sec-dns-server>
                       <pppoe-user/>
                       <mgmt-prof>
                          <type>OntRgMgmtProf</type>
                          <id>
                             <ontrgmgmtprof>1</ontrgmgmtprof>
                          </id>
                       </mgmt-prof>
                       <tr69-eth-svc>
                          <type>EthSvc</type>
                          <id>
                             <ont>2</ont>
                             <ontslot>8</ontslot>
                             <ontethany>1</ontethany>
                             <ethsvc>1</ethsvc>
                          </id>
                       </tr69-eth-svc>
                       <default-wan-svc>
                          <type>EthSvc</type>
                          <id>
                             <ont>2</ont>
                             <ontslot>8</ontslot>
```

```
                                <ontethany>1</ontethany>
                                <ethsvc>1</ethsvc>
                             </id>
                          </default-wan-svc>
                          <disable-on-batt>true</disable-on-batt>
                          <ont>CXNK00205222</ont>
                       </object>
                    </top>
                 </config>
              </edit-config>
           </rpc>
     </soapenv:Body>
</soapenv:Envelope>
```

XML tags and values

The following table lists the XML element tags in a **merge** request for provisioning 844GE Default WAN Service.

For descriptions of common XML element tags, see "Common XML Element Tags for Edit-Config Requests" on page 267.

Element Tag	Data Type	Req'd ?	Description
`<type>`	See descr	Yes	<type> identifies the Remote Gateway (FB) port is the provisioning object type using the following case-sensitive expression - **OntRg**

`<id>`
`<ont>`
`<ontslot>`
`<ontrg>`
	see descr	Yes	These element tags identify the port:
			<ont> identifies the ONT by its E7 scope ID (**1** to **64000000**)
			<ontslot>—The number **8** specifies an RG port
			<ontrg>—RG port number

`<admin>`	see descr		Configure the admin state as **enabled** to enable all traffic on the port
`<subscr-id>`	Char(63		Subscriber ID
`<descr>`	Char(31)		User description

Element Tag	Data Type	Req'd ?	Description
<mgmt-mode>	see descr		The management mode is configured with the case sensitive word **native**. This is the default value when a new ONT is created. Native management mode allows the service provisioning from the E7. The Static and PPPoE parameters are applicable only in this case. The WAN protocol (dhcp, static, pppoe) is applicable only in the native management mode. ---------- The other available management mode is **external**. External management mode allows the RF service provisioning by an external RF configuration file or ACS. **Note:** T-Series ONTs do not support Native Mode as an option for the Management Mode.
<wan-protocol>	see descr		The WAN protocol is configured with the case sensitive word **dhcp.** The other options are **static** and **pppoe**.
<mgmt-prof> <type> <id> <ontrgmgmtprof>	see descr		<type>- identifies the RG Management profile with the case sensitive word **OntRgMgmtProf** <ontrgmgmtprof>- identifies the RG Management profile number
<tr69-eth-svc> <type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr		<type> identifies the TR-69 Eth Service with the case sensitive word **EthSvc**. <ont>—identifies the ONT by its E7 scope ID (1 to 64000000) <ontslot>—The number **8** identifies the port type as an RG port. <ontethany>- identifies the ONT port number <ethsvc> - identifies the data service (1 to 12; typically 1 to 8 for data service).
<tr69-out-tag> <tr69-in-tag>	see descr		Configure the outer and inner tag VLAN ID tags for the access channel (**none**, **1** to **4093).** The content of these tags depends on the management profile used by the RG interface.

Element Tag	Data Type	Req'd ?	Description
<disable-on-batt>	Bool		Identifies the port operational state when the ONT is operating on battery backup power: • **false**—No (default if no tag is provided) • **true—Yes**
<pbit-map> <type> <id> <dscpmap>	see descr		<type> identifies the provisioning object as a DSCP map profile using the following case-sensitive expression: **DscpMap** <dscpmap> identifies the global DSCP Map profile ID (**1** to **10**). Global profile IDs can be viewed in CMS Desktop at the CMS level (**Profile** > **E7/ONT** > **DSCP** > **E7**). **Note:** If these tags are not supplied, **1** (access) is used for the DSCP Map profile.

836GE Residential Services Gateway and 844G GigaCenter ONT Service Provisioning

Creation of an 836GE Residential Services Gateway (RSG) ONT is described in the section "Creating a 836GE RSG or 844G GigaCenter ONT" on page 372.

The section "Creating a Half Bridge (HB) port on an 836GE RSG or 844G GigaCenter ONT" contains the following sub-sections:

- "Provisioning an 836GE RSG or 844G GigaCenter ONT port as an HB port" on page 377
- "Creating a Data service on an 836GE RSG or 844G GigaCenter ONT HB port" on page 379
- "Updating a Data service on an 836GE RSG or 844G GigaCenter ONT HB port" on page 384
- "Deleting a Data service from an 836GE RSG or 844G GigaCenter ONT HB port" on page 388

The section "Creating a RG (Residential Gateway) port on an 836GE RSG or 844G GigaCenter ONT" contains the following sub-sections:

- "Creating a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port" on page 390
- "Deleting a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port" on page 395
- "Updating a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port" on page 397

Note: When a new 836 ONT is created, all four GE ports are provisioned as RG interfaces by default.

Note: The format for the XML request to update (merge) a service on an 836GE RSG or 844G GigaCenter port is identical to the format for a **create** request, with one exception: For the operation attribute in the object tag, use **merge** instead of **create.**

Creating a 836GE RSG or 844G GigaCenter ONT

Sample XML 836GE RSG or 844G GigaCenter ONT creation request

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="37" nodename="NTWK-local" timeout="35000"
username="rootgod" sessionid="7">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>Ont</type>
       <id>
        <ont>101</ont>
       </id>
       <admin>enabled</admin>
       <serno>123456</serno>
       <reg-id></reg-id>
       <subscr-id>111-222-3333</subscr-id>
       <descr>George Flint</descr>
       <ontprof>
        <type>OntProf</type>
        <id>
         <ontprof>148</ontprof>
        </id>
       </ontprof>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>XML input element tags and values
```

XML input element tags

The following table lists the XML element tags in an 836GE RSG or 844G GigaCenter ONT creation request.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description

`<type>`
`<id>`
`<ont>`
	see descr	Yes	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)

`<admin>`	see descr	Yes	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms** **Note:** If this element tag is not supplied in the XML request, the currently-provisioned operational status of the port is used. Calix recommends supplying this element tag to set the intended status.
`<serno>`	Hex(8)	Yes, if <reg-id> not config	Enter one of the following: • **0**—Leave the serial field empty. Hexadecimal representation of the ONT serial number.
`<reg-id>`	Int(10)	Yes, if <serno> not config	ONT registration ID that is the RONTA identifier.
`<subscr-id>`	Char(27)	Yes	ONT subscriber ID (not copied to ONT ports)..
`<descr>`	Char(27)	No	User description

`<ontprof>`
`<type>`
`<id>`
`<ontprof>`
			`<type>` identifies the profile type as ONT using the following case-sensitive expression: **OntProf**
			`<ontprof>` identifies the ID of a pre-defined local ONT profile

Sample XML 836GE RSG or 844G GigaCenter ONT creation response

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="37" nodename="NTWK-local">
   <ok/>
    <data>
     <top>
      <object>
       <type>Ont</type>
       <id>
        <ont>101</ont>
       </id>
       <admin>enabled</admin>
       <ontprof>
        <type>OntProf</type>
        <id>
         <ontprof name="836GE">148</ontprof>
        </id>
       </ontprof>
       <serno>123456</serno>
       <reg-id/>
       <subscr-id>111-222-3333</subscr-id>
       <descr>George Flint</descr>
       <linked-pon/>
       <pwe3prof/>
       <low-rx-opt-pwr-ne-thresh>-30.00</low-rx-opt-pwr-ne-thresh>
       <high-rx-opt-pwr-ne-thresh>-7.00</high-rx-opt-pwr-ne-thresh>
       <us-sdber-rate>5</us-sdber-rate>
      </object>
     </top>
    </data>
   </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML output element tags

The following table lists the XML element tags in a response to create an 836GE RSG or 844G GigaCenter ONT.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Description

`<type>`
`<id>`
`<ont>`
	see descr	`<type>` identifies the provisioning object type using the following case-sensitive expression: **Ont**
		`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)

`<admin>`	see descr	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms**

`<ontprof>`
`<type>`
`<id>`
`<ontprof name="836GE">`
	see desc	`<type>` identifies the profile type as ONT using the following case-sensitive expression: **OntProf**
		`<ontprof>` identifies the ID of the ONT profile

`<serno>`	see desc	Hexadecimal representation of the ONT serial number.
`<reg-id>`	Int(10)	ONT registration ID that is the RONTA identifier.
`<subscr-id>`	Char(27)	ONT subscriber ID
`<descr>`	Char(27)	User description
`<low-rx-opt-pwr-ne-thresh>`	Int	Low threshold for an near end received optical power alarm. The parameters are between -30 and -7.
`<high-rx-opt-pwr-ne-thresh>`		High threshold for an near end received optical power alarm. The parameters are between -30 and -7.

Element Tag	Data Type	Description
<us-sdber-rate>	Int	Threshold of the SDBER (Signal Degraded Bit Error). This parameter is used to set the threshold value above which the bit error rate of the facility constitutes a degraded signal. When the bit error rate exceeds the threshold specified in this parameter, the system raises a signal degraded bit error rate. The SDBER configuration parameters are between 2 and 9, implemented as follows: • 2 (value of 10 to the negative 2) • 3 (value of 10 to the negative 3) • 4 (value of 10 to the negative 4) ….. • 9 (value of 10 to the negative 9) The default value is 5 (value of 10 to the negative 5).

Creating a Half Bridge (HB) port on an 836GE RSG or 844G GigaCenter ONT

The following sections describe how to configure HB ports on an 836GE RSG or 844G GigaCenter ONT:

Provisioning an 836GE RSG or 844G GigaCenter ONT port as an HB port

Sample XML request to provision an 836GE RSG or 844G GigaCenter ONT port as an HB port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
   username="rootgod" sessionid="7">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>OntEthGe</type>
       <id>
        <ont>101</ont>
        <ontslot>3</ontslot>
        <ontethge>2</ontethge>
        </id>
        <intf></intf>
        <admin>enabled-no-alarms</admin>
        <subscr-id>111-222-3333</subscr-id>
        <descr>George Flint</descr>
       </object>
      </top>
     </config>
    </edit-config>
   </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to provision an 836GE RSG or 844G GigaCenter ONT port as an HB port

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to provision an 836GE RSG or 844G GigaCenter ONT port as an HB port

The following table lists the XML element tags in a HB port provisioning creation request on an 836GE RSG or 844G GigaCenter ONT HB port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethge>	see descr	Yes	<type> Identifies the service type as an Ethernet data and video servic on the ONT using the case-sensitive expression **OntEthGe** <ont> Identifies the ONT by its ID (**1** to **64000000**) <ontslot> The number **3** identifies the port type as a Gigabit Ethernet port <ontethge> - Port number
<intf>	see descr	Yes	Leave the XMLl request for the <intf> element empty to provision the ONT port as a HB (half bridge).
<admin>	see descr		Operational status of the AE ONT GE port: • **disabled** • **ienabled** (Th value "ienabled" is used for ONT Ethernet port admin state) • **enabled-no-alarms)** **Note:** If this element tag is not supplied in the XML request, ienabled is used.
<subscr-id>	Char(63)	No	ONT subscriber ID
<descr>	Char(48)	No	ONT description

Creating a Data service on an 836GE RSG or 844G GigaCenter ONT HB port

Sample XML request to create a data service on an 836GE RSG or 844G GigaCenter ONT HB port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>3</ontslot>
        <ontethany>2</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr>George Flint</descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>2</svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
        <id>
         <bwprof>2</bwprof>
        </id>
       </bw-prof>
       <out-tag>none</out-tag>
       <in-tag>none</in-tag>
       <mcast-prof></mcast-prof>
       <pon-cos>derived</pon-cos>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to create a data service on an 836GE RSG or 844G Giga-Center ONT HB port

The attribute name value is returned in the <svctagaction name="">, <bwprof name="">, and <mcastprof name=""> element tags for the data service.

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
    <data>
     <top>
      <object>
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>3</ontslot>
        <ontethany>2</ontethany>
        <ethsvc name="Data1">1</ethsvc>
       </id>
       <admin>enabled</admin>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction name="@exampleDataTagAction" localId="4">2</
          svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
        <id>
         <bwprof name="@DataBwProf" localId="3">2</bwprof>
        </id>
       </bw-prof>
       <descr/>
       <out-tag>none</out-tag>
       <in-tag>none</in-tag>
       <mcast-prof/>
       <pon-cos>derived</pon-cos>
      </object>
     </top>
    </data>
   </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to create a data service on an 836GE RSG or 844G GigaCenter ONT HB port

The following table lists the XML element tags in a data service creation request on an 836GE RSG or 844G GigaCenter ONT HB port.

`operation=`	Char	Yes	Identifies the requested action.
`get-config=`	Bool	Yes	Attribute for requesting configuration parameters in the reply: • "**true**"—Include configuration parameters/element tags in the XML reply. • "**false**"—Do not include configuration parameters/element tags in the XML reply.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
`object operation="create" get-config="true"`	Char	Yes	Identifies the requested action.

`<type>`
`<id>`
`<ont>`
`<ontslot>`
`<ontethany>`
`<ethsvc>`
	see descr	Yes	`<type>` identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—The number **3** identifies the port type as a Gigabit Ethernet port
			`<ontethany>`—ONT Ethernet port number.
			`<ethsvc>`—data or video service number (**1** to **12**).

`<admin>`	see descr	Yes	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms**
`<descr>`	Char(27)	No	User description

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (**1** to **255**).
<bw-prof> <type> <id> <bwprof>	see desc	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<mcast-prof/>	see descr	No	<mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Element Tag	Data Type	Req'd ?	Description
<pon-cos>	see descr	No	Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

Updating a Data service on an 836GE RSG or 844G GigaCenter ONT HB port

The following example shows how to update a data service on an 836 ONT HB port:.

Sample XML request to update a data service on an 836GE RSG or 844G GigaCenter ONT HB port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
   username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>3</ontslot>
        <ontethany>3</ontethany>
        <ethsvc>2</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr>George Flint</descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>2</svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
        <id>
         <bwprof>2</bwprof>
        </id>
       </bw-prof>
       <out-tag>none</out-tag>
       <in-tag>none</in-tag>
       <mcast-prof></mcast-prof>
       <pon-cos>derived</pon-cos>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to update a data service on an 836GE RSG or 844G GigaCenter ONT HB port

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to update a data service on an 836GE RSG or 844G GigaCenter ONT HB port

The following table lists the XML element tags in a data service update request on an 836GE RSG or 844G GigaCenter ONT HB port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
`object operation="merge"`	Char		Identifies the action as a merge request to update service information.

`<type>`
`<id>`
`<ont>`
`<ontslot>`
`<ontethany>`
`<ethsvc>`
	see descr	Yes	`<type>` identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—The number **3** identifies the port type as a Gigabit Ethernet port
			`<ontethany>`—ONT Ethernet port number.
			`<ethsvc>`—data or video service number (**1** to **12**).

`<admin>`	see descr	Yes	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms**
`<descr>`	Char(27)	No	User description

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (**1** to **255**).
<bw-prof> <type> <id> <bwprof>	see desc	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<mcast-prof/>	see descr	No	<mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Element Tag	Data Type	Req'd ?	Description
<pon-cos>	see descr	No	Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

Deleting a Data service from an 836GE RSG or 844G GigaCenter ONT HB port

Sample XML request to delete a data service on an 836GE RSG or 844G GigaCenter ONT HB port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
   username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="delete">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>3</ontslot>
        <ontethany>2</ontethany>
        <ethsvc>1</ethsvc>
       </id>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to delete a data service on an 836GE RSG or 844G GigaCenter ONT HB port

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to delete a data service on an 836GE RSG or 844G GigaCenter ONT HB port

The following table lists the XML element tags required in a **delete** request for a data service on an 836GE RSG or 844G GigaCenter ONT HB port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
`<object operation="delete">`	see desc		Removes the service activation provisioning from the specified port, as well as from the CMS database, and returns a confirmation of the results.

`<type>`
`<id>`
`<ont>`
`<ontslot>`
`<ontethany>`
`<ethsvc>`
	see descr	Yes	`<type>` identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—The number **3** identifies the port type as a Gigabit Ethernet port
			`<ontethany>`—ONT Ethernet port number.
			`<ethsvc>`—data or video service number (**1** to **12**).

Creating a RG (Residential Gateway) port on an 836GE RSG or 844G GigaCenter ONT

The following sections describe how to configure HB ports on an 836GE RSG or 844G GigaCenter ONT:

Creating a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port

Sample XML request to create a data service on an 836GE RSG or 844G GigaCenter ONT RG port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="create" get-config="true">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr>George Flint</descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>3</svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
         <bwprof>2</bwprof>
        </id>
       </bw-prof>
```

```
        <out-tag>none</out-tag>
        <in-tag>none</in-tag>
        <mcast-prof></mcast-prof>
        <pon-cos>derived</pon-cos>
       </object>
      </top>
     </config>
    </edit-config>
   </rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to create a data service on an 836GE RSG or 844G Giga-Center ONT RG port

The attribute name value is returned in the <svctagaction name="">, <bwprof name="">, and <mcastprof name=""> element tags for the data service.

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
   <data>
    <top>
     <object>
      <type>EthSvc</type>
      <id>
       <ont>101</ont>
       <ontslot>8</ontslot>
       <ontethany>1</ontethany>
       <ethsvc name="Data1">1</ethsvc>
      </id>
      <admin>enabled</admin>
      <tag-action>
       <type>SvcTagAction</type>
       <id>
        <svctagaction name="@RSG-TA" localId="3">3</svctagaction>
       </id>
      </tag-action>
      <bw-prof>
       <type>BwProf</type>
       <id>
        <bwprof name="@DataBwProf" localId="3">2</bwprof>
       </id>
      </bw-prof>
      <descr/>
      <out-tag>none</out-tag>
      <in-tag>none</in-tag>
      <mcast-prof/>
      <pon-cos>derived</pon-cos>
     </object>
    </top>
   </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags for a data service creation request on an 836GE RSG or 844G GigaCenter ONT RG port

The following table lists the XML element tags in a data service creation request on an 836GE RSG or 844G GigaCenter ONT RG port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
<type> <id> <ont> <ontslot> <ontethany> <ethsvc>	see descr	Yes	<type> identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc** <ont> identifies the ONT by its E7 scope ID (**1** to **64000000**) <ontslot>—The number **8** identifies the port type as an RG port <ontethany>—ONT Ethernet port number. <ethsvc>—data or video service number (**1** to **12**).
<admin>	see descr	Yes	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms**
<descr>	Char(27)	No	User description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (**1** to **255**).
<bw-prof> <type> <id> <bwprof>	see desc	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).

Element Tag	Data Type	Req'd ?	Description
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<mcast-prof/>	see descr	No	<mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).
<pon-cos>	see descr	No	Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

Deleting a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port

Warning! If you are adding back a data service or are changing an existing service, Calix highly recommends that you modify the existing service rather than delete it. See “Updating a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port” on page 397.

Sample XML request to delete a data service on an 836GE RSG or 844G GigaCenter ONT RG port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
   username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="delete">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to delete a data service on an 836GE RSG or 844G GigaCenter ONT RG port

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to delete a data service on an 836GE RSG or 844G GigaCenter ONT RG port

The following table lists the XML element tags required in a **delete** request for a data service on an 836GE RSG or 844G GigaCenter ONT RG port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description

`<object operation="delete">`
 	
`<object operation="delete" force="true">`
	Force attr is Bool	The force attribute is not required	Removes the service activation provisioning from the specified port, as well as from the CMS database, and returns a confirmation of the results.

`<type>`
`<id>`
`<ont>`
`<ontslot>`
`<ontethany>`
`<ethsvc>`
	see descr	Yes	`<type>` identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc**
			
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			
			`<ontslot>`—The number **8** identifies the port type as an RG port
			
			`<ontethany>`—ONT Ethernet port number.
			
			`<ethsvc>`—data or video service number (**1** to **12**).

Updating a Data service on an 836GE RSG or 844G GigaCenter ONT ONT RG port

The following example shows how to update a data service on an 836 ONT RG port:

Sample XML request to update a data service on an 836GE RSG or 844G GigaCenter ONT RG port

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
 <soapenv:Body>
  <rpc message-id="62" nodename="NTWK-local" timeout="35000"
   username="rootgod" sessionid="8">
   <edit-config>
    <target>
     <running/>
    </target>
    <config>
     <top>
      <object operation="merge">
       <type>EthSvc</type>
       <id>
        <ont>101</ont>
        <ontslot>8</ontslot>
        <ontethany>1</ontethany>
        <ethsvc>1</ethsvc>
       </id>
       <admin>enabled</admin>
       <descr>George Flint</descr>
       <tag-action>
        <type>SvcTagAction</type>
        <id>
         <svctagaction>3</svctagaction>
        </id>
       </tag-action>
       <bw-prof>
        <type>BwProf</type>
        <id>
         <bwprof>2</bwprof>
        </id>
       </bw-prof>
       <out-tag>none</out-tag>
       <in-tag>none</in-tag>
       <mcast-prof></mcast-prof>
       <pon-cos>derived</pon-cos>
      </object>
     </top>
    </config>
   </edit-config>
  </rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML response to update a data service on an 836GE RSG or 844G GigaCenter ONT RG port

```
<soapenv:Envelope>
 <soapenv:Body>
  <rpc-reply message-id="62" nodename="NTWK-local">
   <ok/>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags to update a data service on an 836GE RSG or 844G GigaCenter ONT RG port

The following table lists the XML element tags in a data service update request on an 836GE RSG or 844G GigaCenter ONT RG port.

For descriptions of common E7 XML element tags, see "Common element tags for E7 OS edit-configuration XML Requests" on page 19.

Element Tag	Data Type	Req'd ?	Description
`object operation="merge"`	Char		Identifies the action as a merge request to update service information.

`<type>`
`<id>`
`<ont>`
`<ontslot>`
`<ontethany>`
`<ethsvc>`
	see descr	Yes	`<type>` identifies the service type as an Ethernet data and video service on the ONT using the case-sensitive expression **EthSvc**
			`<ont>` identifies the ONT by its E7 scope ID (**1** to **64000000**)
			`<ontslot>`—The number **3** identifies the port type as a Gigabit Ethernet port
			`<ontethany>`—ONT Ethernet port number.
			`<ethsvc>`—data or video service number (**1** to **12**).

`<admin>`	see descr	Yes	Operational status of the ONT: • **disabled** • **enabled** • **enabled-no-alarms**
`<descr>`	Char(27)	No	User description

Element Tag	Data Type	Req'd ?	Description
<tag-action> <type> <id> <svctagaction>	see descr	Yes	<type> identifies the tag action type as a service tag action using the following case-sensitive expression: **SvcTagAction** <svctagaction> identifies the ID of a pre-defined global service tag action (**1** to **255**).
<bw-prof> <type> <id> <bwprof>	see desc	Yes	<type> identifies the profile type as Ethernet bandwidth using the following case-sensitive expression: **BwProf** <bwprof> identifies the ID of a pre-defined global Ethernet bandwidth profile (**1** to **300**).
<out-tag> <in-tag>	Int	see note	Outer and inner VLAN IDs (2 to 4093, excluding any reserved VLAN IDs). **Note:** If an outer or inner VLAN is not required by the tag action, do not include the element tag in the XML request. If not supplied, the XML reply returns the value **none**.
<mcast-prof/>	see descr	No	<mcastprof> identifies an ID of a pre-defined global multicast profile (**1** to **32**).

Element Tag	Data Type	Req'd ?	Description
<pon-cos>	see descr	No	Class of Service applied to the service: • **derived** is the default behavior for services created with E7 R2.2 or later. • **cos-1** through **cos-4** represents a default, system-defined aggregated CoS for an ONT (BE, AF1, AF2, EF) that are pre-assigned a class of service and the provisioned services are required to have a bandwidth profile that matches the class of service. Bandwidth is assigned as aggregated from the multiple services and mapped to the ONT. If the associated service-tag action was created with a software version earlier than E7 software release R2.2, the values of the selected system-defined cos (1-4) override the associated service-tag parameter selections. • **user-1** through **user-4** represents the PON upstream profiles that specify the traffic class, DBA scheduling priority, and bandwidth limits for the service on the PON port. • **fixed** is the behavior that is the same as a service created in a software version earlier than E7 software release R2.2.

C7 Service Activation and Querying

This section covers the following topics:

Guidelines for C7 Provisioning

When using the examples and explanations in this guide, keep in mind the following:

- The XML NBI supports C7 line cards and services that can be provisioned using the Services screen in CMS Desktop, including pre-provisioned services, except as noted in this list.
- XML service activation requests override any previously provisioned service parameters.
- The XML NBI supports creating C7 ONTs but does not support updating C7 ONTs.
- Using the XML NBI, services on DS1 ports can only be provisioned on ONTs, not on C7 line cards.
- Using the XML NBI, RF return service on C7 ONT RF video ports cannot be provisioned.
- Off Hook status for a line cannot be checked using the XML NBI.
- The C7 facilities and equipment referenced by the XML attributes must support the requested service, including any minimum software release requirements. For specific information refer to the Calix C7 documentation.
- For non-required element tags, if no value is supplied in the request, do not include the element tag in the XML request.
- Calix recommends sending a single C7 service activation request within an XML request whenever possible.

Common XML Element Tags

Common element tags for C7 XML requests

Each C7 XML request contains a SOAP envelope and tags, as shown in the following example excerpt.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         message-id="1444">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
        <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2"
                dependency="false">
              <VideoProvision>
                . . .
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Note: For an example of an additional service request included in the same XML request, see "Provisioning Residential Gateway Service on ADSL and xDSL Ports" on page 460.

Note: For examples of suspend and resume XML requests, see "Suspending and Resuming Service" on page 524.

The following table lists the XML element tags and common attributes in C7 **create**, **query**, **update**, and **delete** requests.

Element Tag or Attribute	Data Type	Req'd ?	Description
`<rpc ... message-id=>`	Positive Integer: 2^31-1		A unique number identifying the request that can be used to match an XML reply with the request.
`<requests module=Msap>`	Char	Yes	This tag is nested under the <netconf:config> tag and identifies the Calix platform. Use the expression to the left in all cases.

Element Tag or Attribute	Data Type	Req'd ?	Description
`<request ...>`	see below	Yes	This tag is nested under the <requests module> tag, and contains the attributes for defining the request type and authenticating the XML request, as described in the next five rows.
`action=`	Char	Yes	Identifies one of the requested actions: • "**create**" • "**update**" • "**query**" • "**delete**" • "**action**" (for ONT resets and suspend/resume requests)
`userName=`	Char	Yes	The name of the user currently logged in to the CMS XML NBI, enclosed n quotes.
`sessionId=`	Int	Yes	Use the session ID returned after logging in to the CMS XML NBI, enclosed in quotes.
`dependency=`	Bool		Optionally include attribute when nesting multiple service requests in the <requests> tag within the same XML request: • **"false"—**Perform requests asynchronously and random in order (default if attribute is not included). • **"true"—**Perform requests one-by-one in order. **Note:** If a requested service activation fails, execution of remaining requests continues but may be affected by the failure.
`*actionName=` * Include only for suspend, resume, and ONT reset requests	Char	See descr	This attribute is only applicable when submitting an **update** action to suspend or resume service, or an **action** action to reset an ONT. Use one of the following values: • "**coldreset**" • "**suspend**" • "**resume**" For more information, see "Resetting C7 ONTs" on page 522 or "Suspending and Resuming Service" on page 524.

Element Tag or Attribute	Data Type	Req'd ?	Description
<DataProvision> <DS0VoiceProvision> <DS1VoiceProvision> <ONTProvision> <SSA> <VideoProvision> < XDSLBondingGroupPort Provision >	tag only	Yes	This tag is nested under the <request> tag, and identifies the service type using the tags listed in the left-most column of this row). Note the following: • Multiple service requests can be nested in the same XML request, for example, when provisioning residential gateway service. • <ONTProvision> is used for create, delete, and reset ONT XML requests. • <SSA> is used for suspend and resume XML requests and audit query request. This chapter describes specific element tags nested under the corresponding tags at the left.

The following descriptions present the values for common element tags used to identify the C7 network, AID, service, and equipment: <NetworkName>, <ID>, <IDType>, and <EqptType>.

<NetworkName> element tag

Case-sensitive name of the C7 network, preceded by NTWK- and enclosed in quotes. Example: **"NTWK-Pet01C7"**

The NetworkName value can consist of alphanumeric, underscore, and space characters.

<ID> element tag AID syntax

Calix C7 access identifier syntax varies by provisioning element type, as follows:

- **ADSL or xDSL port data service:** N{1-255}-{1-5}-{1-20}-{1-24}-{0-1}
  (node, shelf, slot, port, data service 1 [0] or 2 [1])
- **ADSL or xDSL port video* service:** N{1-255}-{1-5}-{1-20}-{1-24}
  (node, shelf, slot, port)
- **ADSL or xDSL bonding group (for bonding group creation, updates, and deletion):**
  N{1-255}-{1-5}-{1-20}-GRP{1-12}
  (node, shelf, slot, bonding group)
- **ADSL or xDSL bonding group data service:**
  N{1-255}-{1-5}-{1-20}-GRP{1-12}-{0-1}
  (node, shelf, slot, bonding group, data service 1 [0] or 2 [1])

- **ADSL or xDSL bonding group video* service:**
  N{1-255}-{1-5}-{1-20}-GRP{1-12}
  (node, shelf, slot, bonding group)
- **DS0 Port (line card):** N{1-255}-{1-5}-{1-20}-{1-24}
  (node, shelf, slot, port)
- **DS0/voice port (ONT):** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-8}
  (node, shelf, OLT slot, PON port, ONT number, voice port)
- **DS1 port (ONT):** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-8}
  (node, shelf, OLT slot, PON port, ONT number, DS1 port)
- **ONT create, delete, reset, and query requests:**
  N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}
  (node, shelf, OLT slot, PON port, ONT number)
- **ONT data service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}-{0-1}
  (node, shelf, OLT slot, PON port, ONT number, Ethernet port,
  data service 1 [0] or 2 [1])
- **ONT video* service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}
  (node, shelf, OLT slot, PON port, ONT number, Ethernet port)

* Note: For suspend and resume service requests (see "Suspending and Resuming Service" on page 524), the AID of the port is used to suspend or resume both video *and* data service.

<IDType> element tag service categories

- **21**—24-port AID for DS0 on C7 line cards and video service on ADSL and xDSL line cards
- **8011**—Port AID for data service on C7 ADSL and xDSL line cards
- **8012**—Port AID for data service on C7 ONTs
- **4944**—AID for create, delete, reset, and query C7 ONT requests
- **4946—**Port AID for DS0, DS1, and video service on C7 ONTs

<EqptType> element tag C7 line card and OLT list

The following list of C7 line cards can be referenced in XML requests (EqptType tag values are case-sensitive):

- **ADSL-24:** CalixADSL24Card
- **ADSL2-24:** CalixADSL2-24Card (Note the dash in this line card name; others have an underscore character.)
- **ADSL2-24A:** CalixADSL2_24ACard
- **Combo-24:** CalixADSLPOTSComboCard

- **Combo2-24:** CalixADSLPOTSCombo2SingleCard
- **Combo2-24A:** CalixCombo2_24ACard
- **Combo2-24D:** CalixADSLPOTSCombo2Card
- **Combo2-24V:** CalixCombo2_24VCard
- **REBS-12:** CalixRebs:12Card
- **RPOTS-24:** CalixRpots:24Card
- **RU2W-24:** CalixRu2w:24Card
- **VDSL2-24:** CalixVDSL2Card

The following list of C7 optical line termination (OLT) cards can be referenced in XML requests:

- **OLTB-2:** CalixOLTB:2Card
- **OLTG-4:** CalixOLTG:4Card
- **OLTG-4E:** CalixOLTG:4ECard

Common element tags for XML replies

The following example shows the common elements of a C7 XML reply.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1444" xmlns="urn:ietf:params:xml:ns:
           netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                 ...
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following table lists the XML element tags and common attributes in replies to C7 action requests.

Element Tag	Data Type	Description
<rpc-reply ...>	Tag/ attribute	This tag is nested under the <Body> tag, and identifies XML parameters and message ID, if used.
message-id=	Positive Integer: 2^31-1	The message ID, as provided in the XML request, enclosed in quotes.
<data>	Tag only	This tag is nested under the <rpc-reply> tag.
<responses ...>	Tag/ attribute	This tag is nested under the <data> tag, and identifies the schema for the platform. For C7, the msap schema is referenced.
<response>	Tag only	This tag is nested under the <responses> tag., and is the parent tag for each individual reply.
<ResultCode>	Int	This tag is nested under the <response> tag, and identifies the result of the requested action: • **0** indicates a successful action. • Example of an action with a reported error: <ResultCode>**609830**</ResultCode>
<ResultMessage/>	Char	This tag is nested under the <response> tag, and identifies the error code (if any). For example: <ResultMessage>**Error occurred while saving provisioning to C7. See Action > List Service Details.**</ResultMessage>
<resultList>	Tag only	This tag is nested under the <response> tag, and is the parent tag for provisioning tag replies.
<DataProvision> <DS0VoiceProvision> <DS1VoiceProvision> <SSA> <VideoProvision> <XDSLBondingGroupPort Provision>	Tag only	These tags are nested under <resultList> tag, and identify the service type (one of the tags listed in the left-most column). The chapter describes the specific element tags nested under each service type tag. **Note:** The <ONTProvision> tag, when used in a request, is not returned.

A reply to a **query** request for data or video service also returns the following synchronization element tag.

Element Tag	Data Type	Description

For data service requests:
`<SynchState>`
 	
For video service requests:
`<VideoActive>`
	Int	Indicates the synchronization status. The response integers indicate one of the following:
		• **0**—The service is not provisioned.
		• **1**—The service is provisioned and consistent with C7 (subscriber data exists in both CMS and C7).
		• **3**—The service is not provisioned (subscriber data does not exist in either CMS or C7).
		• **4**—The service is externally provisioned (subscriber data exists in C7, but not in CMS).

Read (query) Requests

The XML request defines the objects required to retrieve C7 service on ADSL and xDSL ports and ONTs.

This section contains the following topics:

Querying C7 xDSL, Ethernet, DS0, and RF video ports

This section defines the objects required to query C7 ONTs, xDSL and ONT ports, and RF-video ports on C7 ONTs.

Query requests are sent as **query** XML requests with a <SSA> tag nested under the <request> tab. For more information see "Common element tags for C7 XML requests" on page 403 and the examples below.

Sample XML requests

The following is an example of a query request for a C7 Ethernet port:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1503">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="rootgod" sessionId="1001">
              <SSA>
                <ID>N1-1-5-1-5-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>video-data</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

The following is an example of a query request for an ADSL or xDSL port:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1503">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="rootgod" sessionId="1001">
              <SSA>
                <ID>N1-1-5-18</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>video-data</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

The following is an example of a query request for a C7 ONT:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1504">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="rootgod" sessionId="1001">
              <SSA>
                <ID>N1-1-3-3-2</ID>
                <IDType>4944</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>ont</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

XML input element tags and values

The following tables list the XML element tags in a **update** request for suspending and resuming service.

Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of line card, ONT port, or ONT. Refer to "<ID> element tag AID syntax" on page 405 for the required syntax for one of the following: • ONT • ADSL or xDSL port video and data service • ADSL or xDSL bonding group video and data service • DS0 Port (line card) • ONT DS0/voice port • ONT video and data service • ONT RF-video service
<IDType>	Int	Yes	Identifies the port or provisioning object type. Use one of these following values: • **21**—query video, data, and voice services on C7 ADSL and xDSL cards • **4944—**query ONTs • **4946—**query video, data, voice, and RF-video services on C7 ONTs
<NetworkName>	Char(59)	Yes	Identifies the C7 network. See "<NetworkName> element tag" on page 405.
<ServiceType>	Int	Yes	Identifies the type of service to be suspended or resumed using one of the following case-sensitive expressions: • **ont—**C7 ONT • **rf-video—**RF video (C7 ONT only) • **video-data**—video and data service • **voice**—voice service

Sample XML reply

The following is an example of a reply to a query request for a C7 port:

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1503">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <ID>N1-1-5-18</ID>
                <IDType>21</IDType>
                <ServiceType>video-data</ServiceType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <PST>OOS-MA</PST>
                <SST>AINS</SST>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following is an example of a reply to a query request for data service on a C7 ONT:

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1504">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ID>N1-1-3-3-2-1-0</ID>
                <IDType>4946</IDType>
                <ServiceType>video-data</ServiceType>
                <PST>IS-NR</PST>
                <SST>AINS</SST>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
```

```
  <Body>
</Envelope>
```

XML output element tags

The XML element tags in a reply to an **query** request for C7 ports are the same as those used in the request (see "XML input element tags and values" on page 413, with the addition of the following tag.

Element Tags	Data Type	Description
<PST>	Char	Identifies the primary service state of the port or ONT: • **OOS-AU** (out-of-service, autonomous) • **OOS-AUMA** (out-of-service, autonomous and management) • **OOS-MA** (out-of-service, management) • **IS-NR** (in service, normal)
<SST>	Char	Identifies the secondary service state of the port or ONT. For a complete list of secondary service states, refer to the *Calix C7 Provisioning Guide*.

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) data service on ADSL and xDSL ports

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1445">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-1-1-1</ID>
                <IDType>8011</IDType>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
```

```
          </request>
        </requests>
      </netconf:config>
    </netconf:edit-config>
  </netconf:rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for data service on an ADSL or XDSL port include:

- <ID>
- <IDType>
- <EqptType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1445" xmlns="urn:ietf:params:xml:ns:
          netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-3-1-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-DYNE</NetworkName>
                <AidString>N1-1-3-1-0</AidString>
                <DataTemplate>20</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <SynchState>1</SynchState>
                <UplinkVlan>2</UplinkVlan>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </resultList>
          </response>
        </responses>
```

```
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DataProvision> XML tag, the element tags in a reply to a **query** request for data service on an ADSL or xDSL port are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 442), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) video service on ADSL and xDSL ports

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1446">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for video service on an ADSL or xDSL port include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query replies

The following example is a reply to a **query** request for EXA video service on a C7 xDSL port. Service is provisioned and consistent between CMS and C7.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
  <rpc-reply message-id="1447"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <data>
      <responses xmlns="http://schemas.calx.com/nc/msap/">
        <response>
          <ResultCode>0<ResultCode>
          <ResultMessage/>
          <resultList>
            <VideoProvision>
              <ID>N1-1-1-1</ID>
              <IDType>21</IDType>
              <NetworkName>NTWK-XMLNBI</NetworkName>
              <AIDString>N1-1-1-1</AIDString>
              <EqptType>CalixVDSL2Card</EqptType>
              <VideoTemplate>8</VideoTemplate>
              <VideoLuAddressType>4984</VideoLuAddressType>
              <VideoLuAddress>3</VideoLuAddress>
              <Cvid>2</Cvid>
              <RCvid>2</RCvid>
              <Prio>4</Prio>
              <VideoActive>1</VideoActive>
            </VideoProvision>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a reply to a **query** request for video service on a C7 ADSL port. Service is provisioned and consistent between CMS and C7.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1446"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AidString>N1-1-1-1</AidString>
                <EqptType>CalixCombo2_24ACard</EqptType>
                <VideoTemplate>7</VideoTemplate>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <IPBwc>0</IPBwc>
                <VideoActive>1</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following example is a reply to a **query** request for video service on a C7 xDSL port. Service is provisioned and consistent between CMS and C7.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns=”http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
  <rpc-reply message-id="1447"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <data>
      <responses xmlns=”http://schemas.calx.com/nc/msap/”>
        <response>
          <ResultCode>0<ResultCode>
          <ResultMessage/>
          <resultList>
            <VideoProvision>
              <ID>N1-1-1-1</ID>
              <IDType>21</IDType>
              <NetworkName>NTWK-XMLNBI</NetworkName>
              <AIDString>N1-1-1-1</AIDString>
              <ResidentialGatewayEnabled>0</ResidentialGatewayEnabled>
              <VideoTemplate>8</VideoTemplate>
```

```
            <VideoActive>1</VideoActive>
          </VideoProvision>
        </response>
      </responses>
    </data>
  </rpc-reply>
 </soapenv:Body>
</soapenv:Envelope>
```

The following example is a reply to a **query** request for video service on a C7 xDSL port. Service is externally provisioned.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1448">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-3-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <EqptType>CalixAdsl2-24Card</EqptType>
                <VideoActive>4</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to a **query** request for video service on an ADSL or xDSL port are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 451), with these additional element tags:

- <ResidentialGatewayEnabled>
- <VideoActive>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) Residential Gateway service on a C7 ADSL or VDSL port

Sample XML query requests

The following is an example of a **query** Residential Gateway service request for an ADSL port (single data service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         message-id="1449">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-2</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
              <DataProvision>
                <ID>N1-1-1-2-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **query** Residential Gateway service request for an xDSL port with two data services:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1450"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
```

```
          <request action="query" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-15</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
            <request action="query" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-15-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
            <request action="query" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-15-1</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for Residential Gateway service, in each <VideoProvision> and <DataProvision> XML tag, include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query reply

The following is an example of a reply to a **query** request for residential gateway service on an ADSL port. Service is provisioned and consistent between CMS and C7.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1450">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-1-1</AidString>
                <EqptType>CalixADSL2-24Card</EqptType>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
                <VideoTemplate>65</VideoTemplate>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <VideoActive>1</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-1-1-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-1-1-0</AidString>
                <DataTemplate>64</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN4</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <SynchState>1</SynchState>
                <UplinkVlan>4</UplinkVlan>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags (up to two data service responses based on the number of data service queries), the element tags in a reply to a **query** request for Residential Gateway service are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 462), with one additional element tag nested in the <VideoProvision> XML tag:

- <VideoActive>

and one additional element tag nested in each <DataProvision> XML tag:

- <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) DS0 service on a C7 line card

Sample XML query request

The following is an example of a **query** DS0 service request for a C7 line card.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1451"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="rootgod" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-3-13</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-petuxa18</NetworkName>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for DS0 service on a C7 line card include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1451"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-1-IG1</IG>
                <CRV>2</CRV>
                <H248EcMode>0</H248EcMode>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in <DS0VoiceProvision> XML tag, the element tags in a reply to a **query** request for DS0 service on a C7 line card are the same as the input element tags for a **create** request (see "Create DS0 service on C7 line cards" on page 473).

Read (query) data service on a C7 ONT

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1453">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-5-1-1-0</ID>
                <IDType>8012</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for data service on an C7 ONT include <ID>, <IDType>, and <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

Nested in a <DataProvision> XML tag, the element tags in a reply to a **query** request for data service on a C7 ONT are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 485), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) video service on a C7 ONT

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1454">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for video service on a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to a **query** request for video service on a C7 ONT are the same as the input element tags for a **create** request (see "XML input element tags" on page 491), with one additional element tag: <VideoActive>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) Residential Gateway service on a C7 ONT

Sample XML query request

The following is an example of a **query** Residential Gateway service request for a C7 ONT (single data service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1454"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
              <DataProvision>
                <ID>N1-5-1-1-1-0</ID>
                <IDType>8012</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for Residential Gateway service, in each <VideoProvision> and <DataProvision> XML tag, include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags (up to two data service responses based on the number of data service queries), the element tags in a reply to a query request for Residential Gateway service are the same as the input element tags for a create request (see "XML input element tags and values" on page 497), with one additional element tag nested in the <VideoProvision> XML tag:

- <VideoActive>

and one additional element tag nested in each <DataProvision> XML tag:

- <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) DS0 service on a C7 ONT

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1455"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for DS0 service on include:

- <ID>
- <IDType>
- <NetworkName>
- <AIDString> **Note:** Use the same value as the <ID> element tag.

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query replies

The following is an example of a reply to a **query** reply for DS0 H.248 service on a C7 ONT.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1455"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG3</IG>
                <CRV>2</CRV>
                <H248EcMode>1</H248EcMode>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following is an example of a reply to a query reply for DS0 SIP service on a C7 ONT (with DHCP protocol).

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1456"
            xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG4</IG>
                <SipT0UserName>c7</SipT0UserName>
                <SipT0UserPassword>c7</SipT0UserPassword>
                <SipT0UplinkVlan>2</SipT0UplinkVlan>
                <SipT0HostProto>2</SipT0HostProto>
                <SipT0AOR>11</SipT0AOR>
                <RxEthBWProf>12</RxEthBWProf>
                <TxEthBWProf>11</TxEthBWProf>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS0VoiceProvision> XML tag, the element tags in the reply to a **query** request for DS0 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 507).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Read (query) DS1 service on a C7 ONT

Sample XML query request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1457"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="query" userName="cmsuser09" sessionId="2">
              <DS1VoiceProvision>
                <ID>N1-5-1-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-5-1-1-1-1</AIDString>
              </DS1VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **query** request for DS1 service on a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>
- <AIDString> **Note:** Use the same value as the <ID> element tag.

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML query reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1457">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS1VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <Path>5</Path>
                <DataUplinkAddressType>76</DataUplinkAddressType>
                <DataUplinkAddress>N1-1-CSA-1-1</DataUplinkAddress>
                <VP>0</VP>
                <VC>45</VC>
              </DS1VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS1VoiceProvision> XML tag, the element tags in the reply to a **query** request for DS1 service are the same as the element tags in a **create** request (see "XML input element tags and values" on page 517).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Managing xDSL Bonding Groups

The XML request defines the objects required for:

- Adding an xDSL bonding group (see below)
- "Updating members of an xDSL bonding group" on page 438
- "Deleting an xDSL bonding group" on page 439

Configuration process overview

To activate service on an E7 xDSL bonding group, follow these steps:

- Add an xDSL bonding group.
- If not already completed when adding an xDSL bonding group, update the member ports in the bonding group.
- Activate service on the bonding group. See the following:
  - "Create data service on ADSL and xDSL ports and bonding groups" on page 440
  - "Create video service on an ADSL and xDSL ports and bonding groups" on page 450
  - "Create Residential Gateway service on ADSL and xDSL ports" on page 460.

Adding an xDSL bonding group

Sample XML request

The following example show a create request for a bonding group with VDSL2 service.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="144">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="rootgod" sessionId="2">
              <XDSLBondingGroupPortProvision>
                <ID>N1-1-1-GRP1</ID>
                <IDType>18400</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <Members>N1-1-1-1,1-1-1-2</Members>
                <PktMode>1</PktMode>
                <XrDs>65472</XrDs>
                <MrDs>768</MrDs>
                <XrUs>1600</XrUs>
```

```
                <MrUs>256</MrUs>
                <DifDs>16</DifDs>
                <DifUs>16</DifUs>
                <ReptRmvRst>0</ReptRmvRst>
                <GOS>21</GOS>
                <FallbackVpi>0</FallbackVpi>
                <FallbackVci>35</FallbackVci>
                <PromoteAlarms>0</PromoteAlarms>
              </XDSLBondingGroupPortProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example show a create request for a bonding group with ADSL service.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="144">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="rootgod" sessionId="2">
              <XDSLBondingGroupPortProvision>
                <ID>N1-1-1-GRP2</ID>
                <IDType>18400</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <Members>N1-1-1-1,1-1-1-2</Members>
                <MemberPortServiceType>ADSL2+</MemberPortServiceType>
                <PktMode>1</PktMode>
                <XrDs>32000</XrDs>
                <MrDs>768</MrDs>
                <XrUs>1600</XrUs>
                <MrUs>256</MrUs>
                <DifDs>16</DifDs>
                <DifUs>16</DifUs>
                <ReptRmvRst>0</ReptRmvRst>
                <GOS>21</GOS>
                <PromoteAlarms>0</PromoteAlarms>
              </XDSLBondingGroupPortProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for adding an xDSL bonding group.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • N{1-255}-{1-5}-{1-20}-GRP{1-12}—(node, shelf, slot, bonding group)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **18400**—Bonding group AID
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<Members>	Char		xDSL port AIDs (node, shelf, slot, port) delimited by a comma: • N{1-255}-{1-5}-1-20}-{1-24}, N{1-255}-{1-5}-1-20}-{1-24} Example: **N1-1-1-1,N1-1-1-2** **Note:** For rules for member ports within a bonding group, see the Calix C7 documentation.
<MemberPortServiceType>	Char	Required for ADSL service	Identifies the service type for ADSL member ports: • **ADSL2** • **ADSL2+** • **ANNEXM** • **READSL2**
<PktMode>	Int	Required for VDSL2 service	Identifies whether the group operates in packet mode: • **0**—No (default) • **1—**Yes
<XrDs>	Int		Identifies the maximum downstream rate. Default=**65472**.
<MrDs>	Int		Identifies the minimum downstream rate. Default=**32**.
<XrUs>	Int		Identifies the maximum upstream rate. Default=**6144**.

Element Tag	Data Type	Req'd ?	Description
<MrUs>	Int		Identifies the minimum upstream rate. Default=**32**.
<DifDs>	Int		Identifies the allowable downstream differential delay (**0** to **64**). Default=**16**.
<DifUs>	Int		Identifies the allowable upstream differential delay (**0** to **64**). Default=**16**.
<ReptRmvRst>	Int		Identifies whether the group enables reporting of remove/restore events: • **0—Y**es • **1—N**o (default if tag is not provided)
<GOS>	Int		Identifies the GOS profile ID (**1** to **20**), or one of the following: • **21—**Off (default if tag is not provided) • **22**—DEFLT
<PromoteAlarms>	Int		Identifies whether specific bonding group events are promoted to alarms. • **0—**Yes (default if tag is not provided) • **1**—No
<FallbackVpi> **Note:** Only applicable for VDSL2 bonding groups.	Int		VPI value (**0** to **255**). Default = **0**.
<FallbackVci> **Note:** Only applicable for VDSL2 bonding groups.	Int		VCI value (**32** to **255**). Default = **35**.
<Desc>	Char(31)		Identifies the bonding group description.

Sample XML reply

```
<?xml version="1.0" encoding="UTF-8"?>
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
   <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <XDSLBondingGroupPortProvision>
                <ID>N1-1-8-GRP1</ID>
```

```
                <IDType>18400</IDType>
                <NetworkName>NTWK-CMS_S47</NetworkName>
                <Members>N1-1-8-1,N1-1-8-2</Members>
                <PktMode>1</PktMode>
                <XrDs>128000</XrDs>
                <MrDs>768</MrDs>
                <XrUs>51200</XrUs>
                <MrUs>256</MrUs>
                <DifDs>16</DifDs>
                <DifUs>16</DifUs>
                <ReptRmvRst>0</ReptRmvRst>
                <GOS>21</GOS>
                <FallbackVpi>0</FallbackVpi>
                <FallbackVci>35</FallbackVci>
                <PromoteAlarms>0</PromoteAlarms>
              </XDSLBondingGroupPortProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Updating members of an xDSL bonding group

Sample excerpt from XML request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="145">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="rootgod" sessionId="2">
              <XDSLBondingGroupPortProvision>
                <ID>N1-1-1-GRP1<</ID>
                <IDType>18400</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <Members>N1-1-1-3,N1-1-1-4</Members>
                . . .
```

XML input element tags

The element tags required for an **update** request to change the member ports in a bonding group include <ID>, <IDType>, <NetworkName>, and <Members>. For tag descriptions, see the table under "XML input element tags and values" on page 436.

Deleting an xDSL bonding group

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="146">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="rootgod" sessionId="2">
              < XDSLBondingGroupPortProvision >
                <ID>N1-1-1-GRP1</ID>
                <IDType>18400</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </ XDSLBondingGroupPortProvision >
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The element tags required for a **delete** request to change the member ports in a bonding group include <ID>, <IDType>, and <NetworkName>. For tag descriptions, see the table under "XML input element tags and values" on page 436.

Provisioning Data Service on ADSL and xDSL Ports and Bonding Groups

The XML request defines the objects required to configure, retrieve, update, and delete C7 data service on ADSL and xDSL ports and bonding groups.

This section contains the following topics:

For read (query) requests, see "Read (query) data service on ADSL and xDSL ports" on page 415.

Create data service on ADSL and xDSL ports and bonding groups

Sample XML create requests

The following is an example of a **create** data service request for an ADSL port (ATM service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        message-id="1458">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-1-1-1</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <DataTemplate>1</DataTemplate>
                <DataUplinkAddress>N1-1-11-1-1</DataUplinkAddress>
                <DataUplinkAddressType>470</DataUplinkAddressType>
                <VP>0</VP>
                <VC>55</VC>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** data service request for an xDSL port (packet VLAN service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1459">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-13-0</ID>
                <IDType>8011</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>1</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>14</Cvid>
                <RCvid>15</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning data service on an ADSL or xDSL port.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object. Use one of the following: • **ADSL or xDSL port data service:** N{1-255}-{1-5}-{1-20}-{1-24}-{0-1}—(node, shelf, slot, port, data service 1 [0] or 2 [1]) • **ADSL or xDSL bonding group data service:** N{1-255}-{1-5}-{1-20}-GRP{1-12}-{0-1}—(node, shelf, slot, bonding group, data service 1 [0] or 2 [1]).
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **8011**—Port AID for data service on ADSL and xDSL line cards
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	Yes	Identifies the case-sensitive C7 line card type on which services are being provisioned. See "<EqptType> element tag C7 line card and OLT list" on page 406.

Element Tag	Data Type	Req'd ?	Description
<DataTemplate>	Int	Yes	ID number of the global data subscriber template.
<DataUplinkAddress>	Char	Yes	AID of the uplink. Use one of the following formats. • ATM service: N{1-255}-{1-5}-{1-20}-{1-24}-{1-48} (node, shelf, slot, port, STS). Example: **N1-1-11-1-1** • Packet VLAN service: N{1-255}-{1-5}-VB{1-20}-VLAN{2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2** • VLAN index (EXA): VLAN ID from 2 to 4093, excluding reserved VLANs. Example: **4**
<DataUplinkAddressType>	Int	Yes	Identifies the uplink address type: • **20**—Optical (OC) line card • **21**—IMA, ATM Resource, or DS3 port • **469**—STS under OC3 port • **470**—STS under OC12 port • **471**—STS under OC48 port • **4963**—Packet VLAN • **4984**—VLAN Index (EXA)
Additional element tags for ATM service: <UplinkVlan>* <VPI> <VCI> * Applicable for bridged ATM service.	Int	Yes	<UplinkVlan> For ATM services, this tag identifies the VLAN ID for bridged encapsulated cross-connects, and is only applicable for bridged ATM service. <VPI> and <VCI> identify the virtual path identifier and virtual channel identifier.
Additional element tags for Packet VLAN and EXA service: <Cvid> <RCvid> <Prio>	Int		<Cvid>: Customer VLAN ID (2 to 4093, excluding reserved VLANs) <RCvid>: Relay CVID (2 to 4093, excluding reserved VLANs) <Prio>: IEEE 802.1p priority (0 to 7) **Note:** If not provided, traffic is untagged, with priority 0.

Element Tag	Data Type	Req'd ?	Description

Additional element tags for Transparent LAN Service (TLS):
`<Cvid>`
`<RCvid>`
`<Prio>`
	Int	Yes	`<Cvid>`: Use one of the following:
			• Untagged traffic: **1**
			• Default (DFLT) value: **4095**
			`<RCvid>`: Use one of the following:
			• None: **1**
			• Default (DFLT) value: **4095**
			`<Prio>`: Use one of the following:
			• Copy Bits: **8**
			• DSCP: **16**

Sample XML create reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1460"
           xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-3-1-0</ID>
                <IDType>8011</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DYNE</NetworkName>
                <AidString>N1-1-3-1-0</AidString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>20</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <SynchState>1</SynchState>
                <UplinkVlan>2</UplinkVlan>
                <Cvid>14</Cvid>
                <RCvid>111</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DataProvision> XML tag, the element tags in a reply to a **create** request for data service on an ADSL or xDSL port are the same as the input element tags (see "XML input element tags and values" on page 442), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update data service on an ADSL or xDSL port or bonding group

Sample XML update requests

The following is an example of an **update** data service request for an ADSL port (ATM service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1461">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-1-1-1</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <DataTemplate>2</DataTemplate>
                <DataUplinkAddress>N1-1-11-1-4</DataUplinkAddress>
                <DataUplinkAddressType>470</DataUplinkAddressType>
                <VP>0</VP>
                <VC>55</VC>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **update** data service request for an xDSL port (EXA service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1462"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
            <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-13-1</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>11</DataTemplate>
                <DataUplinkAddress>5</DataUplinkAddress>
                <DataUplinkAddressType>4984</DataUplinkAddressType>
                <Cvid>2</Cvid>
                <RCvid>2</RCvid>
                <Prio>2</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags in an **update** data service request for an ADSL or xDSL port are the same as those used in a **create** request (see "XML input element tags and values" on page 442).

For update XML requests, a value for each required element tag must be supplied even if the parameter value is not being changed.

Sample XML update reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1461">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-3-1-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-DYNE</NetworkName>
                <AidString>N1-1-3-1-0</AidString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>22</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <SynchState>1</SynchState>
                <UplinkVlan>2</UplinkVlan>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>2</Prio>
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DataProvision> XML tag, the element tags in a reply to an update request for data service on an ADSL or xDSL port are the same as the input element tags (see "XML input element tags and values" on page 442), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete data service on an ADSL or xDSL port or bonding group

Note: Deleting data service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1471">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
              <DataProvision>
                <ID>N1-1-1-1-1</ID>
                <IDType>8011</IDType>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for data service on an ADSL or xDSL port include:

- <ID>
- <IDType>
- <EqptType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1471">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-3-4-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-DYNE</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for data service on an ADSL or xDSL port include: <ID>, <IDType>, <NetworkName>, and <EqptType> (returned for all except bonding groups). For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning Video Service on ADSL and xDSL Ports and Bonding Groups

The XML request defines the objects required to configure, retrieve, update, and delete video service on C7 ADSL and xDSL ports.

This section contains the following topics:

For read (query) requests, see "Read (query) video service on ADSL and xDSL ports" on page 417.

Create video service on an ADSL and xDSL ports and bonding groups

Sample XML create requests

The following is an example of a **create** video service request on an ADSL port:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1472">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <VideoTemplate>1</VideoTemplate>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <IPBwc>251</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** video service request on an xDSL port:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc message-id="1473"
        xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-13</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <VideoTemplate>1</VideoTemplate>
                <IPBwc>251</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning video service on an ADSL or xDSL port.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object. Use one of the following: • **ADSL or xDSL port video service:** N{1-255}-{1-5}-{1-20}-{1-24}—(node, shelf, slot, port) • **ADSL or xDSL bonding group video service:** N{1-255}-{1-5}-{1-20}-GRP{1-12}—(node, shelf, slot, bonding group)

Element Tag	Data Type	Req'd ?	Description
<IDType>	Int	Yes	Identifies the port or provisioning object type using the following: • **21**—Video service on xDSL (ADSL and VDSL) line cards
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	Yes	Identifies the case-sensitive C7 line card type on which services are being provisioned. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<VideoTemplate>	Int	Yes	ID number of the global video subscriber template.
Additional element tags for EXA video and video service on ADSL ports: <VideoLuAddress> <VideoLuAddressType>	see descr	Yes	<VideoLuAddress> identifies the AID of the EPG channel source for the shelf. Example: N1-1-VB1-VLAN2 (node, shelf, VB, VLAN). <VideoLuAddressType> identifies the EPG channel source type: • **20**—Optical (OC) line card • **21**—IMA, ATM Resource, or DS3 port • **469**—STS under OC3 port • **470**—STS under OC12 port • **471**—STS under OC48 port • **4963**—Packet VLAN • **4984**—VLAN Index
<IPBwc>	Int		Identifies the bandwidth constraint ID.

Sample XML create replies

The following is an example of a reply to a **create** video service request on an ADSL port.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns"urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1472">
      <responses xmlns=”http://schema.calx.com/nc/msap”>
        <data>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AidString>N1-1-1-1</AidString>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <VideoTemplate>7</VideoTemplate>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <IPBwc/>
                <VideoActive>1</VideoActive>
             </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</soapenv:Envelope>
```

The following is an example of a reply to a **create** video service request on an VDSL port.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns"urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1472">
      <responses xmlns=”http://schema.calx.com/nc/msap”>
        <data>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-1-1</AIDString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <VideoTemplate>8</VideoTemplate>
                <IPBwc/>
                <VideoActive>1</VideoActive>
             </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</soapenv:Envelope>
```

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to a create request for video service on an ADSL or xDSL port are the same as the input element tags (see “XML input element tags and values” on page 451), with one additional element tag: <VideoActive>

For descriptions of common XML element tags, see “Common element tags for XML replies” on page 407.

Update video service on ADSL or xDSL port or bonding group

Sample XML update requests

The following is an example of an **update** video service request for an ADSL port.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1474">
      <netconf:edit-config>
        <netconf:target>
            <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <VideoTemplate>2</VideoTemplate>
                <VideoLuAddress>N1-1-11-1-4</VideoLuAddress>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <IPBwc>55</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of an **update** video service request for an xDSL bonding group.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        message-id="1475">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-GRP1</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <VideoTemplate>6</VideoTemplate>
                <IPBwc>251</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in an **update** request for video service on an ADSL or xDSL port are the same as those used in a **create** request (see "XML input element tags and values" on page 451).

For update XML requests, a value for each required element tag must be supplied even if the parameter value is not being changed.

Sample XML update reply

The following is an example of a reply to an **update** video service request for an xDSL bonding group.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns"urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1475">
      <responses xmlns=″http://schema.calx.com/nc/msap″>
        <data>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-3-GRP1</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-3-GRP1</AIDString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <VideoTemplate>6</VideoTemplate>
                <IPBwc>251</IPBwc>
                <VideoActive>1</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</soapenv:Envelope>
```

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to an **update** request for video service on an ADSL or xDSL port are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 451), with one additional element tag: <VideoActive>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete video service on ADSL or xDSL port or bonding group

Note: Deleting video service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1476">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
             <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>8011</IDType>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for video service on an ADSL or xDSL port include:

- <ID>
- <IDType>
- <EqptType>
- <NetworkName>

For descriptions of these element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

The following is an example of a reply to a **delete** video service request.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns"urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1476">
      <responses xmlns=″http://schema.calx.com/nc/msap″>
        <data>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-3-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24VCard</EqptType>
               </VideoProvision>
           </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</soapenv:Envelope>
```

XML output element tags

XML element tags returned from a **delete** request for video service on an ADSL or xDSL port include: <ID>, <IDType>, <NetworkName>, and <EqptType> (returned for all except bonding groups). For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning Residential Gateway Service on ADSL and xDSL Ports

The XML request defines the objects required to configure, retrieve, update, and delete C7 video and data services on ADSL and xDSL ports to support residential gateway service.

This section contains the following topics:

For read (query) requests, see "Read (query) Residential Gateway service on a C7 ADSL or VDSL port" on page 421.

Create Residential Gateway service on ADSL and xDSL ports

Sample XML create requests

The following is an example of a **create** Residential Gateway service request for an ADSL port with video and one data service:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1477">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-2</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <VideoTemplate>11</VideoTemplate>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
```

```
            <request action="create" userName="HeadHoncho2" sessionId="2"
                  dependency="true">
              <DataProvision>
                <ID>N1-1-1-2-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** Residential Gateway service request for an xDSL port with video and two data services:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1478">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-15</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <VideoTemplate>11</VideoTemplate>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2">
                  dependency="true">
              <DataProvision>
                <ID>N1-1-3-15-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
```

```
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2">
               dependency="true">
              <DataProvision>
                <ID>N1-1-3-15-1</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN3</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>2</Cvid>
                <RCvid>2</RCvid>
                <Prio>2</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following tables list the XML element tags in a **create** request for provisioning Residential Gateway service on an ADSL or xDSL port.

Video Provisioning XML input element tags and values

The following table lists the element tags nested under the <VideoProvision> tag.

Video Service Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object. Use one of the following: • **ADSL or xDSL port video service:** N{1-255}-{1-5}-{1-20}-{1-24}—(node, shelf, slot, port) • **ADSL or xDSL bonding group video service:** N{1-255}-{1-5}-{1-20}-GRP{1-12}—(node, shelf, slot, bonding group)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **21**—Video service on xDSL (ADSL and VDSL) line cards
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	Yes	Identifies the case-sensitive C7 line card type on which services are being provisioned. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<VideoTemplate>	Int	Yes	ID number of the global video subscriber template.

Video Service Element Tags	Data Type	Req'd ?	Description

Additional element tags for EXA RG service and RG service on ADSL ports:
`<VideoLuAddress>`
`<VideoLuAddressType>`
	Char	Yes	<VideoLuAddress> identifies the AID of the EPG channel source for the shelf. Example: N1-1-VB1-VLAN2 (node, shelf, VB, VLAN).
			<VideoLuAddressType> identifies the EPG channel source type:
			• **20**—Optical (OC) line card
			• **21**—IMA, ATM Resource, or DS3 port
			• **469**—STS under OC3 port
			• **470**—STS under OC12 port
			• **471**—STS under OC48 port
			• **4963**—Packet VLAN
			• 4**984**—VLAN Index (EXA)

`<ResidentialGateway Enabled>`	Bool	Yes	**1**—enable Residential Gateway service (**0** disables RG service)

Data Provisioning XML input element tags and values

The following table lists the element tags nested under the <DataProvision> tag.

Data Service Element Tags	Data Type	Req'd ?	Description
`<ID>`	Char	Yes	Access Identifier (AID) of the provisioning object. Use one of the following; • **ADSL or xDSL port data service:** N{1-255}-{1-5}-{1-20}-{1-24}-{0-1}—(node, shelf, slot, port, data service 1 [0] or 2 [1]) • **ADSL or xDSL bonding group data service:** N{1-255}-{1-5}-{1-20}-GRP{1-12}-{0-1}—(node, shelf, slot, bonding group, data service 1 [0] or 2 [1])
`<IDType>`	Int	Yes	Identifies the port or provisioning object type: • **8011**—Data service on x (ADSL and VDSL) line cards

Data Service Element Tags	Data Type	Req'd ?	Description
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	Yes	Identifies the case-sensitive C7 line card type on which services are being provisioned. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<DataTemplate>	Int	Yes	ID number of the global data subscriber template.
<DataUplinkAddress>	Char	Yes	AID of the uplink. Use one of the following formats. • ATM service: N{1-255}-{1-5}-{1-20}-{1-24}-{1-48} (node, shelf, slot, port, STS). Example: **N1-1-11-1-1** • Packet VLAN service: N{1-255}-{1-5}-VB{1-20}-VLAN{2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2** • VLAN index (EXA): VLAN ID from 2 to 4093, excluding reserved VLANs. Example: **4**
<DataUplinkAddressType>	Int	Yes	Identifies the uplink address type: • **4963**: Packet VLAN • **4984**: VLAN Index (EXA)

Data Service Element Tags	Data Type	Req'd ?	Description

Additional element tags for Packet VLAN and EXA service:
`<Cvid>`
`<RCvid>`
`<Prio>`
	Int		`<Cvid>`: Customer VLAN ID (2 to 4093, excluding reserved VLANs)
			`<RCvid>`: Relay CVID (2 to 4093, excluding reserved VLANs)
			`<Prio>`: IEEE 802.1p priority (0 to 7)
			Note: If not provided, traffic is untagged, with priority 0.

Sample XML create reply

The following is an example of a reply to a **create** request for residential gateway service for an ADSL port for service that is provisioned and consistent between CMS and C7.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1447">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                 <SubscriberID>Frederick Flinstone</SubscriberID>
                 <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-1-1</AidString>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
                <VideoTemplate>65</VideoTemplate>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <VideoActive>1</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DataProvision>
                <ID>N1-1-1-1-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-1-1-0</AidString>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <DataTemplate>64</DataTemplate>
```

```
                <DataUplinkAddress>N1-1-VB1-VLAN4</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <SynchState>1</SynchState>
                <UplinkVlan>4</UplinkVlan>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags, the element tags in a reply to a create request for Residential Gateway service on an ADSL or xDSL port are the same as the input element tags (see "XML input element tags and values" on page 462), with one additional element tag nested in the <VideoProvision> XML tag:

- <VideoActive>

and one additional element tag nested in each <DataProvision> XML tag:

- <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update Residential Gateway service on an ADSL or xDSL port or bonding group

Sample XML update requests

The following is an example of an **update** Residential Gateway service request for an ADSL port:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1500">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-1-2</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <VideoTemplate>11</VideoTemplate>
                <VideoLuAddress>N1-1-VB1-VLAN3</VideoLuAddress>
                <VideoLuAddressType>4963</VideoLuAddressType>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <DataProvision>
                <ID>N1-1-1-2-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN3</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of an **update** Residential Gateway service request for an xDSL port with two data services.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1501">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="HeadHoncho2" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-15</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24VCard</EqptType>
                <VideoTemplate>11</VideoTemplate>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-15-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24VCard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN4</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <DataProvision>
                <ID>N1-1-3-15-0</ID>
                <IDType>8011</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixADSL2_24VCard</EqptType>
                <DataTemplate>13</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN5</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>2</Cvid>
                <RCvid>2</RCvid>
                <Prio>2</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
```

```
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in an **update** request for video service on an ADSL or xDSL port are the same as those used in a **create** request (see "XML input element tags and values" on page 462).

For required element tags, a value must be supplied even if the parameter value is not being changed.

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags (up to two data service responses based on the number of data service queries), the element tags in a reply to an **update** request for Residential Gateway service are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 462), with one additional element tag nested in the <VideoProvision> XML tag:

- <VideoActive>

and one additional element tag nested in each <DataProvision> XML tag:

- <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete Residential Gateway service on an ADSL or xDSL port or bonding group

Sample XML delete request

Note: Deleting residential gateway service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1502">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-3-15</ID>
                <IDType>21</IDType>
                <EqptType>CalixADSL2_24VCard</EqptType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for Residential Gateway service on an ADSL or xDSL port include:

- <ID>
- <IDType>
- <EqptType>
- <NetworkName>

In the XML API, only the <VideoProvision> tags should be supplied without the <DataProvision> tags. For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1502"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-1-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <EqptType>CalixADSL2_24ACard</EqptType>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for Residential Gateway service on an ADSL or xDSL port include: <ID>, <IDType>, <NetworkName>, and <EqptType> (for all except bonding groups), nested under the <VideoProvision> XML tag. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning DS0 Services on C7 Line Cards

The XML request defines the objects required to configure, retrieve, update, and delete C7 DS0 service on C7 ONTs.

This section contains the following topics:

For read (query) requests, see "Read (query) DS0 service on a C7 line card" on page 424.

Create DS0 service on C7 line cards

Sample XML create request

The following is an example of a **create** DS0 service request for a C7 line card (GR-303 service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1505">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="rootgod" sessionId="2"
               dependency="false">
              <DS0VoiceProvision>
                <ID>N1-1-3-13</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-petuxa18</NetworkName>
                <AidString>N1-1-3-13</AidString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <IG>N1-1-IG2</IG>
                <IGType>1</IGType>
                <CRV>2</CRV>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning DS0 service on a C7 line card. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • N{1-255}-{1-5}-{1-20}-{1-24}—(node, shelf, slot, port)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **21**—DS0 service on C7 line cards
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	Yes	Identifies the case-sensitive C7 line card type on which services are being provisioned. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<AIDString>	Char	Yes	Use the same value as in the <ID> element tag.
<IG>	Char	Yes	Specifies the address (AID) of the interface group used for T0 cross-connects. The AID syntax depends on the interface group type: • For GR-303 or H.248: N{1-255}-{1-6}-IG{1-15} (node, shelf, interface group) • For GR-8 service: N{1-255}-{1-6}-{1-20}-IG{1-15} (node, shelf, slot, interface group)
<IGType>	Int	Yes	Identifies the interface group type: • **1**—GR-303 • **2**—GR-8 • **3**—H.248

Element Tag	Data Type	Req'd ?	Description
Additional element tag for GR-303, GR-8, or H.248 service: <CRV>	Int	Yes	Identifies the call reference value (CRV), channel, or voice signalling processor (VSP): • **GR-303**—The CRV range is 1 to 2048 • **GR-8**—The channel range is 1 to 96 • **H.248**—The VSP is 1 or 2
Additional element tag for H.248 service: <H248EcMode>	Bool		Echo cancellation mode: • **0**—Use the interface group setting (default if no element tag is supplied) • **1**—Off

Sample XML create reply

The following is an example of a **create** DS0 service request for a C7 line card (GR-303 service):

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1505">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1</AidString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <IG>N1-1-IG2</IG>
                <IGType>1</IGType>
                <CRV>2</CRV>
                <IG>N1-1-IG2</IG>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS0VoiceProvision> XML tag, the element tags in the reply to a **create** request for DS0 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 474).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update DS0 service

Sample XML update request

The following is an example of an **update** DS0 service request for an ONT with GR-303 service:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1507">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1</AIDString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <IG>N1-1-IG2</IG>
                <IGType>1</ITType>
                <CRV>3</CRV>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags in an **update** DS0 service request are the same as those used in a **create** request (see "XML input element tags and values" on page 474).

For required element tags, a value must be supplied even if the parameter value is not being changed.

Sample XML update reply

The following is an example of an **update** DS0 service request for a C7 line card (GR-303 service):

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1507">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1</ID>
                <IDType>21</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1</AidString>
                <EqptType>CalixCombo2_24VCard</EqptType>
                <IG>N1-1-IG2</IG>
                <IGType>1</IGType>
                <CRV>9</CRV>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS0VoiceProvision> XML tag, the element tags in the reply to an **update** request for DS0 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 474).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete DS0 service

Note: Deleting DS0 service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1508">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
              <DS0VoiceProvision>
                <ID>N1-1-5-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1</AIDString>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for DS0 service include:

- <ID>
- <IDType>
- <NetworkName>
- <AIDString> **Note:** Use the same value as the <ID> element tag.

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1508">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1</AidString>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for DS0 service include: <ID>, <IDType>, and <NetworkName>. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Creating and Deleting C7 ONTs

The XML request defines the objects required to create and delete C7 ONTs.

This section contains the following topics:

Create C7 ONTs

Sample XML create request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1509">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <ONTProvision>
                <ID>N1-1-5-1-4</ID>
                <IDType>4944</IDType>
                <reg-id>7775551111</reg-id>
                <descr>ont</descr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AdrMode>1</AdrMode>
                <ONTPID>123456</ONTPID>
                <GOS>21</GOS>
                <ONTNum>44</ONTNum>
                <ONTProf>59</ONTProf>
                <SDBER>6</SDBER>
                <VCG>N1-1-IG1</VCG>
                <BatProv>1</BatProv>
              </ONTProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for creating an ONT on a C7 network. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}—(node, shelf, OLT slot, PON port, ONT number)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **4944**—create and delete ONT requests
<reg-ID>	Int		String of up to 10 digits that can be used instead of the <ONTNum> for remote ONT registration.
<desc>	Char(11)		User-defined description of the ONT.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<AdrMode>	Int		Identifies the IP addressing mode for ONT voice services that require an IP interface (SIP or TDM Gateway): • **0**—Group - can only be used by SIP services using VEP model. • **1**—Port - required for SIP services using SIP T0 model, optional for VEP model. If element tag is not supplied, Group (1) is used.
<ONTPID>	Char		Specifies the password used to verify the ONT authenticity.
<GOS>	Int		Grade of Service access identifier. Use the GOS ID (**1** to **20**), or one of the following: • **21**—Off • **22**—Default If the element tag is not supplied, Off is used.
<ONTNum>	see descr		Specifies the serial number of the ONT: • The value can be a decimal or a hexadecimal. • If a hexadecimal is used, it must start with **0x**.

Element Tag	Data Type	Req'd ?	Description
<ONTProf>	Int	Yes	Specifies the defined ONT profile to apply to the ONT. Valid values are **1** to **50**, or one of the following: • **51**—ONT710 • **52**—ONT711 • **53**—ONT712 • **54**—ONT714 • **55**—ONT720 • **56**—ONT721 • **57**—ONT722 • **58**—ONT724 • **59**—ONT740
<SDBER>	Int		Identifies the Signal Degraded Bit Error threshold value n (from **5** to **9**, for 10 to the negative n). If element tag is not supplied, 5 is used.
<VCG>	Char		Specifies the address (AID) of the voice concentration group used to support T0 cross-connects. The AID syntax is N{1-255}-{1-6}-IG{1-15} (node, shelf, interface group), for example **N1-2-IG3**.
<BatProv>	Bool		Indicates whether battery backup capability is expected: • **0**—No • **1**—Yes (default if tag is not supplied)

XML output element tags

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete C7 ONT

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1510">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
             <ONTProvision>
                <ID>N1-1-5-1-4</ID>
                <IDType>4944</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </ONTProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

The XML element tags returned from a **delete** request for a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning Data Service on C7 ONTs

The XML request defines the objects required to configure, retrieve, update, and delete C7 data service on ONT ports.

This section contains the following topics:

For read (query) requests, see "Read (query) data service on a C7 ONT" on page 426.

Create data service on C7 ONTs

Sample XML create request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1511">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-5-1-1-1-0</ID>
                <IDType>8012</IDType>
                <SubscriberID>Frederick Flintone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <DataTemplate>9</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>115</Cvid>
                <RCvid>115</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning data service on a C7 ONT. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **ONT data service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}-{0-1}—(node, shelf, OLT slot, PON port, ONT number, Ethernet port, data service 1 [0] or 2 [1])
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **8012**: Data service on ONTs
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char		Identifies the case-sensitive C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<DataTemplate>	Int	Yes	ID number of the global data subscriber template.

Element Tag	Data Type	Req'd ?	Description
<DataUplinkAddress>	Char	Yes	AID of the uplink. Use one of the following formats. • ATM service: N{1-255}-{1-5}-{1-20}-{1-24}-{1-48} (node, shelf, slot, port, STS). Example: **N1-1-11-1-1** • Packet VLAN service: N{1-255}-{1-5}-VB{1-20}-VLAN{2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2** • VLAN index (EXA): VLAN ID from 2 to 4093, excluding reserved VLANs. Example: **4**
<DataUplinkAddressType>	Int	Yes	Identifies the uplink address type: • **20**—Optical (OC) line card • **21**—IMA, ATM Resource, or DS3 port • **469**—STS under OC3 port • **470**—STS under OC12 port • **471**—STS under OC48 port • **4963**—Packet VLAN • **4984**—VLAN Index (EXA)
Additional element tags for ATM service: <VPI> <VCI>	Int	Yes	VPI and VCI identify the virtual path identifier and virtual channel identifier.
Additional element tags for Packet VLAN and EXA service: <Cvid> <RCvid> <Prio>	Int		<Cvid>: Customer VLAN ID (2 to 4093, excluding reserved VLANs) <RCvid>: Relay CVID (2 to 4093, excluding reserved VLANs) <Prio>: IEEE 802.1p priority (0 to 7). **Note:** If not provided, traffic is untagged, with priority 0.

Element Tag	Data Type	Req'd ?	Description

Additional element tags for Transparent LAN Service (TLS):
`<Cvid>`
`<RCvid>`
`<Prio>`
	Int	Yes	`<Cvid>`: Use one of the following:
			• Untagged traffic: **1**
			• Default (DFLT): **4095**
			`<RCvid>`: Use one of the following:
			• None: **1**
			• Default (DFLT): **4095**
			`<Prio>`: Use one of the following:
			• Copy Bits: **8**
			• DSCP: **16**

XML output element tags

Element tags are nested in a <DataProvision> XML tag. The element tags in a reply to a create request for data service on a C7 ONT are the same as the input element tags (see "XML input element tags and values" on page 485), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update data service on a C7 ONT

Sample XML update request

The following is an example of an **update** data service request for a C7 ONT (EXA service).

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1512">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-5-1-1-1-1</ID>
                <IDType>8012</IDType>
```

```
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <DataTemplate>11</DataTemplate>
                <DataUplinkAddress>4</DataUplinkAddress>
                <DataUplinkAddressType>4984</DataUplinkAddressType>
                <Cvid>2</Cvid>
                <RCvid>2</RCvid>
                <Prio>2</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags in an **update** data service request for a C7 ONT are the same as those used in a **create** request (see "XML input element tags and values" on page 485).

For required element tags, a value must be supplied even if the parameter value is not being changed.

XML output element tags

Nested in a <DataProvision> XML tag, the element tags in a reply to an update request for data service on a C7 ONT are the same as the input element tags for a create request (see "XML input element tags and values" on page 485), with one additional element tag: <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete data service on a C7 ONT

Note: Deleting data service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1513">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
              <DataProvision>
                <ID>N1-1-5-1-1-1-0</ID>
                <IDType>8012</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for data service on a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

The XML element tags returned from a **delete** request for data service on a C7 ONT include: <ID>, <IDType>, and <NetworkName>. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning Video Service on C7 ONTs

The XML request defines the objects required to configure, retrieve, update, and delete video service on C7 ONTs.

This section contains the following topics:

For read (query) requests, see "Read (query) video service on a C7 ONT" on page 427.

Create video service on a C7 ONT

Sample XML create request

The following is an example of a **create** EXA video service request on a C7 ONT.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1513">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <VideoLuAddressType>4984</VideoLuAddressType>
                <VideoLuAddress>101</VideoLuAddress>
                <VideoTemplate>9</VideoTemplate>
                <IPBwc>251</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags in a **create** request for provisioning ONT video service. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **ONT video service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}—(node, shelf, OLT slot, PON port, ONT number, Ethernet port) Example: **N1-1-4-2-1-1**
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **4946**—Video service on C7 ONTs
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char		Identifies the case-sensitive C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<VideoTemplate>	Int	Yes	ID number of the global video subscriber template.

Element Tag	Data Type	Req'd ?	Description

Additional element tags required for EXA video service
`<VideoLuAddress>`
`<VideoLuAddressType>`
	Char	Yes	<VideoLuAddress> identifies the AID of the EPG channel source for the shelf. N{1-255}-{1-5}-VB{1-20}-VLAN{2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2**
			<VideoLuAddressType> identifies the EPG channel source type:
			• **4963**—Packet VLAN
			• **4984**—VLAN Index (EXA)

`<IPBwc>`	Int		Identifies the bandwidth constraint ID.

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to a create request for video service on a C7 ONT are the same as the input element tags, with one additional element tag: <VideoActive>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update video service on a C7 ONT

Sample XML update requests

The following is an example of an **update** video service request for a C7 ONT.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1514">
      <netconf:edit-config>
        <netconf:target>
            <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
```

```
                <EqptType>CalixOLTG:4Card</EqptType>
                <VideoTemplate>10</VideoTemplate>
                <IPBwc>55</IPBwc>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in an **update** request for video service on a C7 ONT are the same as those used in a **create** request (see "XML input element tags" on page 491).

For required element tags, a value must be supplied even if the parameter value is not being changed.

Sample XML update reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1514"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <VideoTemplate>30</VideoTemplate>
                <IPBwc/>
                <VideoActive>1</VideoActive>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <VideoProvision> XML tag, the element tags in a reply to an **update** request for video service on a C7 ONT are the same as the input element tags for a **create** request (see "XML input element tags" on page 491), with one additional element tag: <VideoActive>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete video service on a C7 ONT

Note: Deleting video service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1515">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
             <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for video service on a C7 ONT include:

- <ID>
- <IDType>
- <NetworkName>

For descriptions of these element tags, see "Common element tags for C7 XML requests" on page 403.

XML output element tags

The XML element tags returned from a **delete** request for video service on a C7 ONT include: <ID>, <IDType>, and <NetworkName>. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning Residential Gateway Service on C7 ONTs

The XML request defines the objects required to configure, update, and delete C7 video and data services on C7 ONTs to support residential gateway service.

This section contains the following topics:

For read (query) requests, see "Read (query) Residential Gateway service on a C7 ONT" on page 428.

Create Residential Gateway service on C7 ONTs

Sample XML create request

The following is an example of a **create** Residential Gateway service request for a C7 ONT with video and one data service:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       message-id="1516">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="HeadHoncho2" sessionId="2">
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <VideoTemplate>9</VideoTemplate>
                <VideoLuAddress>N1-1-VB1-VLAN2</VideoLuAddress>
                <VideoLuAddressType>4984</VideoLuAddressType>
                <IPBwc>251</IPBwc>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
            <request action="create" userName="HeadHoncho2" sessionId="2"
                  dependency="true">
```

```
            <DataProvision>
              <ID>N1-1-5-1-1-1-0</ID>
              <IDType>8012</IDType>
              <NetworkName>NTWK-XMLNBI</NetworkName>
              <EqptType>CalixOLTG:4Card</EqptType>
              <DataTemplate>9</DataTemplate>
              <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
              <DataUplinkAddressType>4963</DataUplinkAddressType>
              <Cvid>1</Cvid>
              <RCvid>1</RCvid>
              <Prio>0</Prio>
            </DataProvision>
          </request>
        </requests>
      </netconf:config>
    </netconf:edit-config>
  </netconf:rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following tables list the XML element tags in a **create** request for provisioning Residential Gateway service on a C7 ONT. The first table lists the element tags nested under the <VideoProvision> tag. The second table lists the element tags nested under the <DataProvision> tab.

Video Service Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **ONT video service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}—(node, shelf, OLT slot, PON port, ONT number, Ethernet port)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **4946**—Video service on C7 ONTs
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.

Video Service Element Tags	Data Type	Req'd ?	Description
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char		Identifies the case-sensitive C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<VideoTemplate>	Int	Yes	ID number of the global video subscriber template.
Additional element tags required for EXA RG service <VideoLuAddress> <VideoLuAddressType>	Char	Yes	<VideoLuAddress> identifies the AID of the EPG channel source for the shelf: N{1-255}-{1-5}-VB{1-20}-VLAN{2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2** <VideoLuAddressType> identifies the EPG channel source type: • **4963**—Packet VLAN • **4984**—VLAN Index (EXA)
<IPBwc>	Int		Identifies the bandwidth constraint ID.
<ResidentialGateway Enabled>	Int	Yes	Include a value of **1** to enable Residential Gateway service.

Data Service Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **ONT data service:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-12}-{0-1}— (node, shelf, OLT slot, PON port, ONT number, Ethernet port, data service 1 [0] or 2 [1])
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **8012**—Data service on ONTs

Data Service Element Tags	Data Type	Req'd ?	Description
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted. **Note:** The Subscriber ID and User Description for the provisioning record are the same for data and video services. When supplied, the last saved values under the <DataProvision> or <VideoProvision> tag are used.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted. See the note above for <SubscriberID>.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char		Identifies the case-sensitive C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<DataTemplate>	Int	Yes	ID number of the global data subscriber template.
<DataUplinkAddress>	Char	Yes	AID of the uplink. Use one of the following formats. • Packet VLAN service: N{1-255}-{1-5}-VB{1-20}-VLAN {2-4093} (node, shelf, VB, VLAN). Example: **N1-1-VB1-VLAN2** • VLAN index (EXA): VLAN ID from 2 to 4093, excluding reserved VLANs. Example: **4**
<DataUplinkAddressType>	Int	Yes	Identifies the uplink address type: • **4963**—Packet VLAN • **4984**—VLAN Index (EXA)
<Cvid> <RCvid> <Prio>	Int		<Cvid>: Customer VLAN ID (2 to 4093, excluding reserved VLANs) <RCvid>: Relay CVID (2 to 4093, excluding reserved VLANs) <Prio>: IEEE 802.1p priority (0 to 7) **Note:** If not provided, traffic is untagged, with priority 0.

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags for one or two data services, the element tags in a reply to a create request for Residential Gateway service on an ADSL or xDSL port are the same as the input element tags (see "XML input element tags and values" on page 497), with one additional element tag nested in the <VideoProvision> XML tag:

- <VideoActive>

and one additional element tag nested in each <DataProvision> XML tag:

- <SynchState>

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update Residential Gateway service on C7 ONTs

Sample XML update request

The following is an example of an **update** Residential Gateway service request for a C7 ONT:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1516">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <VideoTemplate>9</VideoTemplate>
                <IPBwc>251</IPBwc>
                <ResidentialGatewayEnabled>1</ResidentialGatewayEnabled>
              </VideoProvision>
            </request>
```

```
            <request action="create" userName="HeadHoncho2" sessionId="2"
                  dependency="true">
              <DataProvision>
                <ID>N1-1-5-1-1-1-0</ID>
                <IDType>8012</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
                <DataTemplate>9</DataTemplate>
                <DataUplinkAddress>N1-1-VB1-VLAN2</DataUplinkAddress>
                <DataUplinkAddressType>4963</DataUplinkAddressType>
                <Cvid>1</Cvid>
                <RCvid>1</RCvid>
                <Prio>0</Prio>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in an **update** request for video service on an ADSL or xDSL port are the same as those used in a **create** request (see "XML input element tags and values" on page 497).

For required element tags, a value must be supplied even if the parameter value is not being changed.

XML output element tags

Nested in <VideoProvision> and <DataProvision> XML tags (up to two data service responses based on the number of data service queries), the element tags in a reply to an **update** request for Residential Gateway service are the same as the input element tags for a **create** request (see "XML input element tags and values" on page 497), with one additional element tag nested in the <VideoProvision> XML tag: <VideoActive>, and one additional element tag nested in each <DataProvision> XML tag: <SynchState>.

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete Residential Gateway service on C7 ONTs

When deleting residential gateway service, only send the video service deletion request. The data service will automatically be deleted.

Note: Deleting residential gateway service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1518">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2">
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
              </VideoProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for Residential Gateway service on an ADSL or xDSL port include:

- <ID>
- <IDType>
- <EqptType>
- <NetworkName>

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1518"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <VideoProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <EqptType>CalixOLTG:4Card</EqptType>
              </VideoProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for Residential Gateway service on an ADSL or xDSL port include: <ID>, <IDType>, and <NetworkName>. These tags are nested in <VideoProvision> XML tag. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning DS0 Services on C7 ONTs

The XML request defines the objects required to configure, retrieve, update, and delete C7 DS0 service on C7 ONTs.

This section contains the following topics:

For read (query) requests, see "Read (query) DS0 service on a C7 ONT" on page 429.

Create DS0 service on C7 ONTs

Sample XML create requests

The following is an example of a **create** DS0 service request for an ONT (GR-303 service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1519">
       <netconf:edit-config>
            <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
           <request action="create" userName="cmsuser09" sessionId="2">
             <DS0VoiceProvision>
               <ID>N1-1-5-1-1-1</ID>
               <IDType>4946</IDType>
               <SubscriberID>Frederick Flinstone</SubscriberID>
               <UserDescr>Bedrock Apts.</UserDescr>
               <NetworkName>NTWK-XMLNBI</NetworkName>
               <AidString>N1-1-5-1-1-1</AidString>
               <EqptType>CalixOLTG:4Card</EqptType>
               <IG>N1-1-IG2</IG>
               <IGType>1</IGType>
               <CRV>2</CRV>
             </DS0VoiceProvision>
           </request>
         </requests>
       </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** DS0 service request for an ONT (H.248 service):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1520">
      <netconf:edit-config>
         <netconf:target>
           <netconf:url/>
         </netconf:target>
         <netconf:config>
           <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG3</IG>
                <IGType>3</IGType>
                <CRV>2</CRV>
                <H248EcMode>1</H248EcMode>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** DS0 service request for an ONT (SIP T0 service with DHCP host protocol):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1521">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
```

```
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG4</IG>
                <IGType>4</IGType>
                <SipT0UserName>123</SipT0UserName>
                <SipT0UserPassword>123</SipT0UserPassword>
                <SipT0UplinkVlan>2</SipT0UplinkVlan>
                <SipT0UplinkVlanType>4984</SipT0UplinkVlanType>
                <SipT0HostProto>2</SipT0HostProto>
                <SipT0AOR>7075551111</SipT0AOR>
                <TxEthBWProf>101</TxEthBWProf>
                <RxEthBWProf>101</RxEthBWProf>
                <VlanIfTemplate>9</VlanIfTemplate>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **create** DS0 service request for an ONT (SIP TDM Gateway service with DHCP host protocol):

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1522">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG5</IG>
                <IGType>5</IGType>
                <SipT0UplinkVlan>2</SipT0UplinkVlan>
                <SipT0UplinkVlanType>4984</SipT0UplinkVlanType>
                <SipT0HostProto>2</SipT0HostProto>
                <SipT0AOR>N1-1-IG5-42</SipT0AOR>
                <TxEthBWProf>101</TxEthBWProf>
```

```
                <RxEthBWProf>101</RxEthBWProf>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning DS0 service on a C7 ONT. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **DS0/voice port (ONT):** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-8}—(node, shelf, OLT slot, PON port, ONT number, voice port)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **4946**—DS0 service on C7 ONTs
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<EqptType>	Char	See descr	Identifies the C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406. This tag is only required for SIP service with an OLT-B OLT card.
<AIDString>	Char	Yes	Use the same value as in the <ID> element tag.

Element Tag	Data Type	Req'd ?	Description
<IG>	Char	Yes	Specifies the address (AID) of the interface group used for T0 cross-connects. The AID syntax depends on the interface group type: • For GR-303, H.248, SIP T0 or TDM Gateway service: N{1-255}-{1-6}-IG{1-15} (node, shelf, interface group) • For GR-8 service: N{1-255}-{1-6}-{1-20}-IG{1-15} (node, shelf, slot, interface group)
<IGType>	Int	Yes	Identifies the interface group type: • **1**—GR-303 • **2**—GR-8 • **3**—H.248 • **4**—SIP T0 • **5**—TDM Gateway
Additional element tag for GR-303, GR-8, or H.248 service: <CRV>	Int	Yes	Identifies the call reference value (CRV), channel, or voice signaling processor (VSP): • GR-303: CRV range is 1 to 2048 • GR-8: Channel range is 1 to 96 • H.248: VSP range is 1 to 2
Additional element tag for H.248 service: <H248EcMode>	Boolean		Echo Cancellation Mode: • **0**—Use the interface group setting (default if no element tag is supplied) • **1**—Off

Element Tag	Data Type	Req'd ?	Description

Additional element tags for SIP T0 service:
`<SipT0UserName>`
`<SipT0UserPassword>`
`<SipT0UplinkVlan>`
`<SipT0UplinkVlanType>`
`<SipT0HostProto>`
`*<SipT0IpAddress>`
`*<SipT0IpMask>`
`*<SipT0IpGateway>`
`*<SipT0VirtualRouter>`
`<SipT0AOR>`
`<TxEthBWProf>`
`<RxEthBWProf>`
`**<VlanIfTemplate>`
 	
* Element tags required for SIP T0 service with static host protocol
** Element tag applicable when a VLAN interface (VLAN-IF) is being created
	Varies	Varies	`<SipT0UserName>`: User name for subscriber (up to 20 bytes in length)
			`<SipT0UserPassword>`: Password for subscriber activation (up to 20 bytes in length)
			`<SipT0UplinkVlan>` (required): Uplink VLAN ID
			`<SipT0UplinkVlanType>` (required): Uplink VLAN ID type:
			• **4693**—Packet VLAN
			• **4984**—VLAN index (EXA)
			`<SipT0HostProto>`: SIP T0 Host protocol:
			• **2**—DHCP (default if tag is not supplied)
			• **3**—Static
			• **6**—Group
			`*<SipT0IpAddress>`: SIP host IP address
			`*<SipT0IpMask>`: SIP host IP subnet mask
			`*<SipT0IpGateway>`: IP address of gateway interface router
			`*<SipT0VirtualRouter>`: Virtual router for static IP host record
			`<SipT0AOR>` (required): Address of Record or URI
			`<TxEthBWProf>` and `<RxEthBWProf>` Transmit and Receive Ethernet bandwidth profiles:
			• **0**—None (default if no element tag is supplied)
			• **101**:—ONTVEPLINE
			• **102**—ONTVEPLINE3
			• A pre-defined profile name
			`<VlanIfTemplate>`: VLAN-IF template ID. (1 to 20).

Element Tag	Data Type	Req'd ?	Description

Additional element tags for SIP TDM Gateway service:
`<SipT0UplinkVlan>`
`<SipT0UplinkVlanType>`
`<SipT0HostProto>`
`<SipT0AOR>`
`<TxEthBWProf>`
`<RxEthBWProf>`
`*<VlanIfTemplate>`
 	
* Element tag applicable when a VLAN interface (VLAN-IF) is being created
	Varies	
		
		Yes
		Yes
		Yes
		
		see descr
			`<SipT0UplinkVlan>`: Uplink VLAN ID
			`<SipT0UplinkVlanType>`: Uplink VLAN ID type:
			• **4693—**Packet VLAN
			• **4984**—VLAN index (EXA)
			`<SipT0HostProto>`: SIP T0 Host protocol:
			• **2**—DHCP (default if tag is not supplied)
			• **3—**Static
			`<SipT0AOR>`: Address of Record or URI
			`<TxEthBWProf>` and `<RxEthBWProf>`: Transmit and Receive Ethernet bandwidth profiles:
			• **0**—None (default if element tag is not supplied)
			• **101**—ONTVEPLINE
			• **102**—ONTVEPLINE3
			• A pre-defined profile name
			`<VlanIfTemplate>`: VLAN-IF template ID. (**1** to **20**).

Sample XML create reply

The following is an example of a **create** DS0 service request for a C7 ONT (SIP service with DHCP protocol):

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1521"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-16-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
```

```
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG1</IG>
                <IGType>4</IGType>
                <SipT0UserName>123</SipT0UserName>
                <SipT0UserPassword>123</SipT0UserPassword>
                <SipT0UplinkVlan>2</SipT0UplinkVlan>
                <SipT0HostProto>2</SipT0HostProto>
                <SipT0AOR>11</SipT0AOR>
                <RxEthBWProf>12</RxEthBWProf>
                <TxEthBWProf>11</TxEthBWProf>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS0VoiceProvision> XML tag, the element tags in the reply to a **create** request for DS0 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 507).

Note: The <VlanIfTemplate> tag, if provided in the request for SIP T0 or TDM Gateway service, is not returned in the reply since it is not stored as part of the voice provisioning record.

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update DS0 service

Sample XML update request

The following is an example of an **update** DS0 service request for an ONT with GR-303 service:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1525">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG2</IG>
                <IGType>1</ITType>
                <CRV>3</CRV>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags in an **update** DS0 service request are the same as those used in a **create** request (see "XML input element tags and values" on page 507).

For required element tags, a value must be supplied even if the parameter value is not being changed.

Sample XML update reply

The following is an example of an **update** DS0 service request for a C7 ONT (H.248 service):

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1526"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <IG>N1-1-IG3</IG>
                <IGType>3</IGType>
                <CRV>9</CRV>
                <H248EcMode>1</H248EcMode>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS0VoiceProvision> XML tag, the element tags in the reply to an **update** request for DS0 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 507).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete DS0 service

Note: Deleting DS0 service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1527">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
              </DS0VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for DS0 service include:

- <ID>
- <IDType>
- <NetworkName>
- <AIDString> **Note:** Use the same value as the <ID> element tag.

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1527"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS0VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <AidString>N1-1-5-1-1-1</AidString>
              </DS0VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for DS0 service include: <ID>, <IDType>, and <NetworkName>. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Provisioning DS1 Services on C7 ONTs

The XML request defines the objects required to configure, retrieve, update, and delete C7 DS1 service on ONTs.

This section contains the following topics:

For read (query) requests, see "Read (query) DS1 service on a C7 ONT" on page 432.

Create DS1 service

Sample XML create request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1528">
      <netconf:edit-config>
        <netconf:target>
            <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="create" userName="cmsuser09" sessionId="2">
              <DS1VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <DataUplinkAddress>N1-1-11-1-1</DataUplinkAddress>
                <DataUplinkAddressType>470</DataUplinkAddressType>
                <Path>5</Path>
                <DataBwc>251</DataBwc>
                <VP>0</VP>
                <VC>55</VC>
              </DS1VoiceProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **create** request for provisioning DS1 service on a C7 ONT. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **DS1 port (ONT):** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}-{1-8}—(node, shelf, OLT slot, PON port, ONT number, DS1 port)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • **4946**—Port AID for DS1 service on ONTs.
<SubscriberID>	Char(32)		Subscriber ID for the port. Alphanumeric, spaces, and special characters are permitted.
<UserDescr>	Char(32)		User description for the port. Alphanumeric, spaces, and special characters are permitted.
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.
<AIDString>	Char	Yes	Use the same value as in the <ID> element tag.
<EqptType>	Char		Identifies C7 OLT line card. See "<EqptType> element tag C7 line card and OLT list" on page 406.
<DataUplinkAddress>	Char	Yes	AID of the uplink. Examples: • ATM service: N1-1-11-1-1 (Node, shelf, slot, port, STS) • Packet VLAN service: N1-1-VB1-VLAN2 (Node, shelf, VB, VLAN) • VLAN index (EXA): 4
<DataUplinkAddress Type>	Int	Yes	Identifies the uplink address type: • **20**—Optical (OC) line card • **21**—IMA, ATM Resource, or DS3 port • **469**—STS under OC3 port • **470**—STS under OC12 port • **471**—STS under OC48 port

Element Tag	Data Type	Req'd ?	Description
<Path>	Int	Yes	Indicates the path type for the cross-connection: • **4**—Both working and protect • **5**—Unprotected (default if no element tag is supplied)
<DataBwc>	Int		Specify an pre-defined bandwidth constraint ID to route traffic over a predefined path through the network.
Additional element tags for ATM service: <VPI> <VCI>	Int	Yes	VPI and VCI identify the virtual path identifier and virtual channel identifier.

Sample XML create reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1528"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS1VoiceProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Frederick Flinstone</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <Path>5</Path>
                <DataUplinkAddress>N1-1-11-1-1</DataUplinkAddress>
                <DataUplinkAddressType>470</DataUplinkAddressType>
                <DataBwc>251<DataBwc/>
                <VP>0</VP>
                <VC>55</VC>
              </DS1VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

Nested in a <DS1VoiceProvision> XML tag, the element tags in the reply to a **create** request for DS1 service are the same as the element tags in a **create** request (see "XML input element tags and values" on page 517).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Update DS1 service

Sample XML update request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1529">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="cmsuser09" sessionId="2">
              <DataProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <SubscriberID>Barney Rubble</SubscriberID>
                <UserDescr>Bedrock Apts.</UserDescr>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
                <EqptType>CalixOLTG:4Card</EqptType>
                <DataUplinkAddress>N1-1-11-1-1</DataUplinkAddress>
                <DataUplinkAddressType>470</DataUplinkAddressType>
                <Path>5</Path>
                <DataBwc>251</DataBwc>
                <VP>0</VP>
                <VC>55</VC>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags in an **update** DS0 service request for a C7 ONT are the same as those used in a **create** request (see "XML input element tags and values" on page 517).

For required element tags, a value must be supplied even if the parameter value is not being changed.

XML output element tags

Nested in a <DS1VoiceProvision> XML tag, the element tags in the reply to an **update** request for DS1 service are the same as the element tags in a **create** request ("XML input element tags and values" on page 517).

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Delete DS1 service

Note: Deleting DS1 service also removes the Subscriber ID (<SubscriberID>) and User Description (<UserDescr>) from the port, if provisioned.

Sample XML delete request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1530">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="delete" userName="cmsuser09" sessionId="2"
              <DataProvision>
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
                <AIDString>N1-1-5-1-1-1</AIDString>
              </DataProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The XML element tags required in a **delete** request for DS1 service include:

- <ID>
- <IDType>
- <NetworkName>
- <AIDString> **Note:** Use the same value as the <ID> element tag.

For descriptions of these XML element tags, see "Common element tags for C7 XML requests" on page 403.

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply message-id="1530"
          xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <DS1VoiceProvision>f
                <ID>N1-1-5-1-1-1</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </DS1VoiceProvision>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

The XML element tags returned from a **delete** request for DS1 service include: <ID>, <IDType>, and <NetworkName>. For descriptions of these XML element tags, see "Common element tags for XML replies" on page 407.

Resetting C7 ONTs

The XML request defines the objects required to reset C7 ONTs.

Reset C7 ONTs

Sample XML reset request

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1509">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="action" userName="cmsuser09" sessionId="2"
                   actionNameType="coldreset">
              <ONTProvision>
                <ID>N1-1-6-4-1</ID>
                <IDType></IDType>
                <NetworkName>NTWK-XMLNBI</NetworkName>
              </ONTProvision>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags and values

The following table lists the XML element tags in a **action** request for resetting an ONT on a C7 network. For descriptions of common XML element tags, see "Common element tags for C7 XML requests" on page 403.

Element Tag	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of the provisioning object: • **ONT reset requests:** N{1-255}-{1-5}-{1-20}-{1-4}-{1-64}— (node, shelf, OLT slot, PON port, ONT number)
<IDType>	Int	Yes	Identifies the port or provisioning object type: • —ONT requests
<NetworkName>	Char(59)	Yes	See "<NetworkName> element tag" on page 405.

XML output element tags

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

Suspending and Resuming Service

This section defines the objects required to suspend and resume C7 ONTs, services (data, video, and voice) on C7 ADSL and xDSL cards and C7 ONTs, and RF-video on C7 ONTs. For ADSL and xDSL cards, service can be suspended and resumed for a single port or for a bonding group.

Suspending services places the Primary Service State on the port to OOS and the Secondary Service State to SB. Resuming services places the Primary Service State on the port to IS (equipment must be present) and the Secondary Service State to SB-DEA.

Before suspending or resuming service on an RF-Video port on a C7 ONT, the port must be created using the management interface (Calix Management System [CMS] or C7 iMS). By default the system only creates an AVO port when the ONT is added.

Suspend and resume requests are sent as **action** XML requests with an additional actionName attribute in the <request> tag and an <SSA> tag nested under the <request> tab. For more information see "Common element tags for C7 XML requests" on page 403 and the examples below.

Sample XML suspend and resume service requests

The following is an example of a request to suspend a C7 ONT:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1503">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="action" userName="rootgod" sessionId="1001"
                 actionName="suspend">
              <SSA>
                <ID>N1-1-5-1-5</ID>
                <IDType>4944</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>ont</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

The following is an example of a request to suspend video and data service on an ADSL or xDSL port:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1503">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="action" userName="rootgod" sessionId="1001"
                 actionName="suspend">
              <SSA>
                <ID>N1-1-5-18</ID>
                <IDType>21</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>video-data</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

The following is an example of a request to resume data service on a C7 ONT:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          message-id="1504">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="action" userName="rootgod" sessionId="1001"
                 actionName="resume">
              <SSA>
                <ID>N1-1-3-3-2-1-0</ID>
                <IDType>4946</IDType>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ServiceType>video-data</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

XML input element tags and values

The following tables list the XML element tags in a **update** request for suspending and resuming service.

Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Access Identifier (AID) of line card, ONT port, or ONT. Refer to "<ID> element tag AID syntax" on page 405 for the required syntax for one of the following: • ONT • ADSL or xDSL port video and data service • ADSL or xDSL bonding group video and data service • DS0 Port (line card) • ONT DS0/voice port • ONT video and data service • ONT RF-video service
<IDType>	Int	Yes	Identifies the port or provisioning object type. Use one of these following values: • **21**—suspend and resume actions on video, data, and voice services on C7 ADSL and xDSL cards • **4944—**suspend and resume actions on ONTs • **4946—**suspend and resume actions on video, data, voice, and RF-video services on C7 ONTs
<NetworkName>	Char(59)	Yes	Identifies the C7 network. See "<NetworkName> element tag" on page 405.
<ServiceType>	Int	Yes	Identifies the type of service to be suspended or resumed using one of the following case-sensitive expressions: • **ont—**C7 ONT suspend/resume • **rf-video—**RF video (C7 ONT only) suspend/resume • **video-data**—video and data service suspend/resume • **voice**—voice service suspend/resume

Sample XML suspend reply

The following is an example of a reply to a suspend request video and data services on a C7 ADSL port:

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1503">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ID>N1-1-5-18</ID>
                <IDType>21</IDType>
                <ServiceType>video-data</ServiceType>
                <PST>OOS-MA, SB</PST>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following is an example of a reply to a suspend request for a C7 ONT:

```
<?xml version="1.0" encoding="UTF-8"?>
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
   <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <ONTProvision>
                <ID>N3-1-7-4-1</ID>
                <IDType>4944</IDType>
                <NetworkName>NTWK-DreamLand</NetworkName>
                <AdrMode>0</AdrMode>
                <AlmProf>6</AlmProf>
                <GOS>21</GOS>
                <ONTNum>10793</ONTNum>
                <SDBER>5</SDBER>
                <ONTProf>54</ONTProf>
                <BatProv>1</BatProv>
```

```
            </ONTProvision>
          </resultList>
        </response>
      </responses>
    </data>
  </rpc-reply>
 </Body>
</Envelope>
```

The following is an example of a reply to an **action** request to resume data service on a C7 ONT:

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1504">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msap">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-DREAM</NetworkName>
                <ID>N1-1-3-3-2-1-0</ID>
                <IDType>4946</IDType>
                <ServiceType>video-data</ServiceType>
                <PST>IS-NR</PST>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  <Body>
</Envelope>
```

XML output element tags

The XML element tags in a reply to an **action** request for suspending and resuming service are the same as those used in the request (see "XML input element tags and values" on page 526, with the addition of the following tags.

Element Tags in ONT Replies	Data Type	Description
<AdrMode>	Int	Identifies the IP addressing mode for ONT voice services with an IP interface (SIP or TDM Gateway): • **0**—Group - used by SIP services using the VEP model. • **1**—Port - used by SIP services using the SIP T0 model, optional for VEP model.

Element Tags in ONT Replies	Data Type	Description
<AlmProf>	Int	Specifies the alarm profile.
<GOS>	Int	Grade of Service ID. Valid values are **1** to **22**.
<ONTNum>	see descr	Specifies the serial number of the ONT: • The value can be a decimal or a hexadecimal. • If a hexadecimal is used, it must start with **0x**.
<SDBER>	Int	Identifies the Signal Degraded Bit Error threshold value n (from **5** to **9**, for 10 to the negative n).
<ONTProf>	Int	Specifies the ONT profile applied to the ONT. Valid values are **1** to **59**.
<BatProv>	Bool	Indicates whether battery backup capability is expected: • **0**—No • **1**—Yes

Element Tags in all Replies	Data Type	Description
<PST>	Char	Identifies the primary service state of the port or ONT using one of the following case-sensitive expressions: • **OOS-AUMA, SB** (out-of-service, suspend both rx and tx, fault conditions present) • **OOS-MA, SB** (out-of-service, suspend both rx and tx, no fault conditions 'present) • **IS-NR** (in service, normal)
<SST>	Char	Identifies the secondary service state of the port or ONT. For a complete list of secondary service states, refer to the *Calix C7 Provisioning Guide*.

For descriptions of common XML element tags, see "Common element tags for XML replies" on page 407.

E3 and E5-100 Service Activation and Querying

This section presents the following information:

Guidelines for E3 and E5-100 Provisioning

When using the examples and explanations in this guide, keep in mind the following:

- All E3 and E5-100 services that can be provisioned using the Services screen in CMS Desktop are supported using the XML NBI.
- While multiple service requests can be nested in the same XML request, Calix recommends sending one service request per XML request, when possible.
- The CMS R12.0 XML NBI does not support configuring the following:
  - Transparent LAN service (TLS)
  - SRA xDSL port settings
  - Bonding groups with a single port on E3-48 units
- XML service activation requests override any previously provisioned service parameters.

Common XML Element Tags

Common element tags for E3/E5-100 XML requests

Each E3/E5 XML service activation request contains a SOAP envelope and tags, as shown in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1601">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="HeadHoncho2" sessionId="2"
                dependency="true">
              <SSA>
                <NetworkName>NTWK-E5111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision>
                  . . .
                </SSAProvision>
            <request action="create" userName="HeadHoncho2" sessionId="2"
                dependency="false">
              <SSA>
                <NetworkName>NTWK-E5111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <PortSetupProvision>
                  . . .
                </PortSetupProvision>
                <VideoProvision>
                  . . .
                </VideoProvision>
                <DataProvision>
                  . . .
                </DataProvision>
              </SSA>
[           <request action="create" userName="HeadHoncho2" sessionId="2"
                dependency="false">
              <SSA>
                . . .
              </SSA>
            </request>]
          </requests>
        </netconf:config>
      </netconf:edit-config>
```

```
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

For examples of suspend and resume XML requests, see "E3-48 Service Activation" on page 576.

For examples of read XML requests, see "XML read Requests" on page 540.

Note: While multiple provisioning requests can be nested in the same XML request, Calix recommends sending a service activation request for a single port.

The following table lists the XML element tags that are common in E3/E5-100 **read**, **create**, **update**, and **delete** requests.

Element Tag	Data Type	Req'd ?	Description
`<rpc ... message-id=>`	Positive Integer: 2^31-1		A unique number identifying the request, used to match an XML reply with the request.
`<edit-config>`	N/A	Yes	This tag is nested under <rpc> tag, and identifies the request as an edit-configuration type.
`<requests module="E5100">`	see descr	Yes	This tag is nested under <config> tag, and identifies the request as an E3/E5-100 service unit type. Use the expression to the left in all cases.
`<request ...>`	N/A	Yes	This tag is nested under the <requests module> tag, and contains the attributes for defining the XML request, as described in the next five rows. **Important:** Separate <request> tags must be included for subscriber information and port setup/provisioning. See the above example to understand how these can be structured within the XML request.
`action=`	Char	Yes	Identifies the requested action: • "**read**" • "**create**" • "**update**" • "**delete**"
`user name=`	Char	Yes	The name of the user currently logged in to the CMS XML NBI, in quotes: Example: "**HeadHoncho2**"

Element Tag	Data Type	Req'd ?	Description
`session ID=`	Int	Yes	Use the session ID returned after logging in to the CMS XML NBI, in quotes: Example: "**2**"
`dependency=`	Bool		Include the dependency attribute when multiple service requests are nested in the <requests module> tag within the same XML request: • **"false"**—Perform requests asynchronously and in random order (default if attribute is not included). • **"true"**—Perform requests one-by-one in order. **Note:** If a requested service activation fails, execution of remaining requests continues but may be affected by the failure.
`*actionName=` * Include only for suspend and resume requests	Char	See descr	This attribute is only applicable when submitting an **update** action to suspend or resume service. Use one of the following values: • **"suspend"** • **"resume"** For more information, see "E3-48 Service Activation" on page 576

For read, create, and delete requests, and for update requests other than suspend and resume requests*:
`<NetworkName>`
`<PortNumber>`
`<PortType>`
 	
* For element tags used with update requests to suspend or resume service, see "E3-48 Service Activation" on page 576
	See descr	Yes	These tags are nested under the <SSA> tag, and identify the network name, port numbers, and port type. Valid values for these tags are as follows:
			<NetworkName> identifies the case-sensitive name of the E3/E5-100 service unit, preceded by "NTWK-". Example: "NTWK-Pet03_E5-121". The value can consist of up to 59 alphanumeric, underscore, and space characters.
			<PortNumber>*—
			E3-48 and E5-110/111: 1 to 48
			E5-120/121: 1 to 24
			E3-12C: 1 to 12
			* For bonding groups, use the master port number.
			<PortType>—
			E5-110/111: **adsl**
			E3-12C, E3-48, E5-120/121: **vdsl**

Element Tag	Data Type	Req'd ?	Description

For create and delete requests, and for update requests other than suspend and resume requests:
`<SSAProvision>`
 	
`<PortSetupProvision>*`
`<DataProvision>`
`<VideoProvision>`
`<VoiceProvision>`
 	
* Not used in **delete** requests
	N/A	See descr	These tags are nested under the <SSA> tag, and identify the port setup parameters and service types (data, video, and voice). Note the following;
			• The <SSAProvision> tag must be included in a separate <request> tag, as shown in the above example.
			• For **create** requests, the <PortSetupProvision> tag is required in all cases. Supply one or more of the service type tags based on the services being provisioned.
			• This chapter describes specific element tags nested under the tags at the left.

For update requests to suspend and resume services: `<SSA>`	N/A	See descr	These tags are nested under the <SSA> tag, and identify the port setup parameters and service types (data, video, and voice). Note the following; • For **create** requests, the <PortSetupProvision> tag is required in all cases. Supply one or more of the service type tags based on the services being provisioned. • This chapter describes specific element tags nested under the tags at the left.

Common XML element tags for E3/E5-100 replies

Each E3/E5 XML reply contains a SOAP envelope and tags, as shown in the following example excerpt showing the results of a successful provisioning request.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1601">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  ...
                </SSAProvision>
              </SSA>
            </resultList>
          </response>
        </responses>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  ...
                  <PortSetupActive>1</PortSetupActive>
                </PortSetupProvision>
                <VideoProvision>
                  ...
                  <VideoActive>1</VideoActive>
                </VideoProvision>
                <DataProvision>
                  ...
                  <DataActive>1</DataActive>
                </DataProvision>
                <VoiceProvision>
                  ...
                  <VoiceActive>1</VoiceActive>
                </VoiceProvision>
              </SSA>
            </resultList>
          </response>
```

```
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following example excerpt shows the results of an unsuccessful provisioning request.

```
Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1602">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>589824</ResultCode>
            <ResultMessage>All local profile names are null, please make sure
              all profiles in the data template are synchronized to the
              E5 device.</ResultMessage>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E52341</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                . . .
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

The following table lists the XML element tags that are common in E3/E5 replies.

Element Tag	Data Type	Description
<rpc-reply ...>	see descr	This tag is nested under <soapenv:Body> tag, and identifies the message ID, as described in the next row.
message-id=	Positive Integer: 2^31-1	The message ID provided in the XML request, enclosed in quotes.
<ResultCode> <ResultMessage/>	Int or string	**0** is returned in the <ResultCode> tag on a successful request. A nonzero value is returned if there is an error message, and the message is included in the <ResultMessage> tag. These tags are nested under the <responses> tag.

Element Tag	Data Type	Description
<resultList> <SSA>	Tag only	These tags contain the provisioning element tags and are nested under the <response> tag. The provisioning element tags are nested under the <SSA> tag.
<PortSetupProvisioning> <PortSetupActive>	Int	For XML replies to **read**, **create**, and **update** requests, one of the following values is returned for each of the Active tags: • **0**—Inconsistent with CMS—Inconsistent provisioning has been detected between the CMS database and the port. • **1**—Provisioned with CMS—Service on the port has been provisioned at one time using CMS. • **3**—Not Provisioned with CMS—Either the service was not provisioned using the CMS management interface, or port settings have been cleared from the CMS database. **Note:** Typically 1 (Provisioned with CMS) is returned for a create or update request.
<VideoProvisioning> <VideoActive>	Int	
<DataProvisioning> <DataActive>	Int	
<VoiceProvisioning> <VoiceActive>	Int	

XML read Requests

This section contains the following topics:

Sample XML read requests

The following is an example of a **read** request for an E5-110 port.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1603">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="read" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_110</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision/>
              </SSA>
            </request>
            <request action="read" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_110</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **read** request for an E5-111 port.

```
<?xml version="1.0"
 encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1604">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <NetworkName>NTWK-E5_111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision/>
              </SSA>
            </request>
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <NetworkName>NTWK-E5_111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
                <VoiceProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **read** request for an E5-120 port.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1605">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <NetworkName>NTWK-E5_120CMS</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision/>
              </SSA>
            </request>
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <NetworkName>NTWK-E5_120CMS</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following is an example of a **read** request for an E5-121 port.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1606">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <SSAProvision/>
              </SSA>
            </request>
            <request action="read" userName="rootgod" sessionId="2"
               dependency="false">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
                <VoiceProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags for **read** requests.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>	Int	Yes	Identifies the port number. See "Common element tags for E3/E5-100 XML requests" on page 533.

Element Tag	Data Type	Req'd ?	Description
<PortType>	see descr	Yes	Identifies the port type. See "Common element tags for E3/E5-100 XML requests" on page 533.
<SSAProvision/>	N/A		Include this tag in the left column (in a separate <request> tag to include port subscriber information in the reply.
<PortSetupProvision/> <VideoProvision/> <DataProvision/> <VoiceProvision/>* * Applies to E3-12C, E5-111, and E5-121 units.	N/A		Include one or more of the tags in the left column to include the corresponding provisioning data in the reply.

Sample XML read reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1606">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Frank Smith</UserDescr>
                  <SubscriberID>555-111-2222</SubscriberID>
                </SSAProvision>
              </SSA>
            </resultList>
          </response>
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
```

```
                    <DSLProfile>DEFVAL</DSLProfile>
                    <BondingStatus>0</BondingStatus>
                    <ResidentialGateway>0</ResidentialGateway>
                    <IPQosProfile>DEFVAL</IPQosProfile>
                    <DSInp>6</DSInp>
                    <DSPhyR>0</DSPhyR>
                    <USInp>6</USInp>
                    <DSPhyR>0</DSPhyR>
                    <PortSetupActive>1</PortSetupActive>
                  </PortSetupProvision>
                  <VideoProvision>
                    <VideoTemplate>2</VideoTemplate>
                    <VideoVLAN>5</VideoVLAN>
                    <VideoActive>1</VideoActive>
                  </VideoProvision>
                  <DataProvision>
                    <DataTemplate>2</DataTemplate>
                    <DataVLAN>6</DataVLAN>
                    <DoubleTagging>0</DoubleTagging>
                    <DataActive>1</DataActive>
                  </DataProvision>
                  <VoiceProvision>
                    <VoIPNumber>98765432</VoIPNumber>
                    <VoiceTemplate>1</VoiceTemplate>
                    <VoiceActive>1</VoiceActive>
                  </VoiceProvision>
                </SSA>
              </resultList>
            </response>
          </responses>
        </data>
      </rpc-reply>
    </Body>
</Envelope>
```

XML output element tags

For descriptions of common XML element tags, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

Based on the requested parameters in the XML request, the element tags for port setup, video, data, and voice service provisioning are in the reply. For descriptions of provisioning parameters, see the element tag tables in the following sections.

E5-110 and E5-111 Service Activation

Sample XML create requests

Sample XML create request for video and data

The following example shows an XML create request without bonding, residential gateway mode, or double tagging.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1610">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
          <request action="create" userName="rootgod" sessionId="2">
            <SSA>
              <NetworkName>NTWK-E5110</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <SSAProvision>
                <UserDescr>Frank Smith</UserDescr>
                <SubscriberID>555-111-2222</SubscriberID>
              </SSAProvision>
            </SSA>
          </request>
          <request action="create" userName="rootgod" sessionId="2">
            <SSA>
              <NetworkName>NTWK-E5110</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <PortSetupProvision>
                <DSLMode>4</DSLMode>
                <DSLProfile>DEFVAL</DSLProfile>
                <BondingStatus>0</BondingStatus>
                <ResidentialGateway>0</ResidentialGateway>
              </PortSetupProvision>
              <VideoProvision>
                <VideoTemplate>1</VideoTemplate>
                <VideoVLAN>100</VideoVLAN>
              </VideoProvision>
              <DataProvision>
                <DataTemplate>1</DataTemplate>
                <DataVLAN>101</DataVLAN>
                <DoubleTagging>0</DoubleTagging>
              </DataProvision>
            </SSA>
```

```
          </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML create request with bonding and residential gateway mode enabled, and with double tagging of data traffic.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1611">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
          <request action="create" userName="rootgod" sessionId="2">
            <SSA>
              <NetworkName>NTWK-E5110</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <SSAProvision>
                <UserDescr>Frank Smith</UserDescr>
                <SubscriberID>555-111-2222</SubscriberID>
              </SSAProvision>
            </SSA>
          </request>
          <request action="create" userName="rootgod" sessionId="2">
            <SSA>
              <NetworkName>NTWK-E5110</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <PortSetupProvision>
                <DSLMode>4</DSLMode>
                <DSLProfile>DEFVAL</DSLProfile>
                <BondingStatus>1</BondingStatus>
                <ResidentialGateway>1</ResidentialGateway>
              </PortSetupProvision>
              <DataProvision>
                <DataTemplate>1</DataTemplate>
                <DataVLAN>101</DataVLAN>
                <DoubleTagging>1</DoubleTagging>
                <RCVid>101</RCVid>
                <Prio>4</Prio>
                <RCPrio>4</RCPrio>
              </DataProvision>
            </SSA>
```

```
          </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML create request for video, data, and SIP voice service (E5-111 port)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1612">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
          <request action="create" userName="rootgod" sessionId="2"
               dependency="true">
            <SSA>
              <NetworkName>NTWK-E5111</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <SSAProvision>
                <UserDescr>Frank Smith</UserDescr>
                <SubscriberID>555-111-2222</SubscriberID>
              </SSAProvision>
            </SSA>
          </request>
          <request action="create" userName="rootgod" sessionId="2"
               dependency="true">
            <SSA>
              <NetworkName>NTWK-E5111</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <PortSetupProvision>
                <DSLMode>4</DSLMode>
                <DSLProfile>DEFVAL</DSLProfile>
                <BondingStatus>0</BondingStatus>
                <ResidentialGateway>0</ResidentialGateway>
              </PortSetupProvision>
              <VideoProvision>
                <VideoTemplate>1</VideoTemplate>
                <VideoVLAN>100</VideoVLAN>
              </VideoProvision>
              <DataProvision>
                <DataTemplate>1</DataTemplate>
                <DataVLAN>101</DataVLAN>
```

```
            <DoubleTagging>0</DoubleTagging>
          </DataProvision>
          <VoiceProvision>
            <!-- SIP Mode Start -->
            <VoIPNumber>9876543210</VoIPNumber>
            <VoiceTemplate>1</VoiceTemplate>
            <!-- SIP Mode End -->
          </VoiceProvision>
        </SSA>
      </request>
      </requests>
    </netconf:config>
  </netconf:edit-config>
 </netconf:rpc>
</soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags for **create** requests. As shown in the above examples, a separate <request> tag must be supplied to request the parameters under the <SSAProvision> tag.

Element Tag	Data Type	Req'd ?	Description
`<NetworkName>`	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
`<PortNumber>*` * For bonding pairs, use the odd-numbered master port.	Int	Yes	Identifies the port. See "Common element tags for E3/E5-100 XML requests" on page 533.
`<PortType>`	`See descr`	Yes	Identifies the port type. Use the following case-sensitive expression: **adsl**

`<SSAProvision>`
`<UserDescr>`
`<SubscriberID>`
	
	Char(31)
	Char(15)
		See descr	Following company policy and procedures, include these tags in a separate <request> tag to add a description and subscriber ID to the record:
			`<UserDescr>`—user-defined field for subscriber information. Spaces and dashes are permitted.
			`<SubscriberID>`—user-defined field for the subscriber ID, such as a telephone number. Spaces and dashes are permitted.

Element Tag	Data Type	Req'd ?	Description
<PortSetupProvision> <DSLMode> <DSLProfile> <BondingStatus> <ResidentialGateway>* **Notes:** * When residential gateway mode is enabled, the <VideoProvision> tags are not supplied; the template used for the <DataTemplate> tag must include the multicast VLAN (MVLAN).	Int Char Bool Bool	Yes	<DSLMode> identifies the ADSL mode using one of the following: • **1**—G.Lite • **2**—G.DMT • **3**—T1.413 • **4**—Auto • **6**—ADSL2 • **7**—ADSL2+ <DSLProfile> identifies the ADSL profile name (up to 31 characters, excluding spaces and dashes). Profile names prefixed with the @ symbol designate global CMS profiles. <BondingStatus> identifies whether bonding is enabled (**1**) or disabled (**0**) for the port. See the note above under <PortNumber> regarding master ports. <ResidentialGateway> identifies whether residential gateway mode is enabled (**1**) or disabled (**0**) for the port. See the note to the left.
<VideoProvision> <VideoTemplate> <VideoVLAN>	Int Int	see descr	<VideoTemplate> identifies the video subscriber template (**1** to **500**). <VideoVLAN> identifies the video service VLAN (**1** to **4094**). **Note:** When residential gateway mode is enabled, do not include these tags.

Element Tag	Data Type	Req'd ?	Description
<DataProvision> <DataTemplate> <DataVLAN> <DoubleTagging> <RCVid>* <Prio>* <RCPrio>* * Tags required when double tagging is enabled.	 Int Int Bool Int Int Int	For data and RG service	<DataTemplate> identifies the data subscriber template (**1** to **500**). When residential gateway mode is enabled, the data template used specifies the multicast VLAN (MVLAN). <DataVLAN> identifies the data service VLAN (**1** to **4094**). <DoubleTagging> identifies whether double tagging is turned on (**1**) or off (**0**) for the port. When enabled, provide the additional fields: <RCVid>: CVID (**1** to **4094**) <Prio>: SVID Priority bit (**0** to **7**) <RCPrio> CVID Priority bit (**0** to **7**)
E5-111 SIP VoIP service: <VoiceProvision> <VoIPNumber> <VoiceTemplate> <VoIPUserName>* <VoIPPassword>* **Note:** * When <VoIPUserName> and <VoIPPassword> are supplied, a new local call services profile is created using the attributes of the global Call Services profile specified in voice subscriber template. The <VoIPUserName> value is used as the new call services profile name as well as the port SIP authorization user name.	 Char Int Char Char	For SIP voice service (see descr)	<VoIPNumber> identifies the user ID (up to 15 characters) as the SIP user name (for URI type = SIP) or telephone number (for URI type = Tel). <VoiceTemplate> identifies the voice subscriber template (**1** to **500**). <VoIPUserName> optionally identifies the user name to use for authentication. See the note to the left. <VoIPPassword> optionally identifies the password to use for authentication. See the note to the left.

Element Tag	Data Type	Req'd ?	Description

E5-111 C7 TDM Gateway VoIP service:
`<VoiceProvision>`
`<VoIPNumber>`
`<VoiceTemplate>`
`<CSIPNetworkName>`
`<CSIPVCG>`
	
	`Char`
	`Int`
	`Char`
	`Char`
		For C7 TDM Gateway voice service	`<VoIPNumber>` identifies the case-sensitive CRV or channel ID for the subscriber line, as provisioned on the C7 GR-303 or GR-8 interface group. Example: **N1-1-IG1-224**
			Tip: Use a voice template that has a pre-defined TDM service group to prevent duplicate CRVs across service units.
			`<VoiceTemplate>` identifies the voice subscriber template (**1** to **500**).
			`<CSIPNetworkName>` identifies the case-sensitive name of the C7 network providing TDM gateway service, preceded by NTWK-. Example: **NTWK-Pet01_C7**
			`<CSIPVCG>` identifies the case-sensitive AID of the C7 SIP voice concentration group. Example: **N1-1-IG2**

E5-111 H.248 service:
`<VoiceProvision>`
`<VoiceTemplate>`
	
	`Int`
		For H.248 voice service	`<VoiceTemplate>` identifies the voice subscriber template (**1** to **500**).

Replies to a create request

Sample XML create reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1612">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision>
                  <UserDescr>Frank Smith</UserDescr>
                  <SubscriberID>555-111-2222</SubscriberID>
```

```
            </SSAProvision>
          </SSA>
        </resultList>
      </response>
      <response>
        <ResultCode>0</ResultCode>
        <ResultMessage/>
        <resultList>
          <SSA>
            <NetworkName>NTWK-E5_111</NetworkName>
            <PortNumber>1</PortNumber>
            <PortType>adsl</PortType>
            <PortSetupProvision>
              <DSLMode>4</DSLMode>
              <DSLProfile>DEFVAL</DSLProfile>
              <BondingStatus>0</BondingStatus>
              <ResidentialGateway>0</ResidentialGateway>
              <PortSetupActive>1</PortSetupActive>
              <PortSetupErrMsg/>
            </PortSetupProvision>
            <VideoProvision>
              <VideoTemplate>1</VideoTemplate>
              <VideoVLAN>100</VideoVLAN>
              <VideoActive>1</VideoActive>
            </VideoProvision>
            <DataProvision>
              <DataTemplate>1</DataTemplate>
              <DataVLAN>101</DataVLAN>
              <DoubleTagging>0</DoubleTagging>
              <DataActive>1</DataActive>
            </DataProvision>
            <VoiceProvision>
              <VoIPNumber>9876543210</VoIPNumber>
              <VoiceTemplate>1</VoiceTemplate>
              <VoiceActive>1</VoiceActive>
            </VoiceProvision>
          </SSA>
        </resultList>
      </response>
    </responses>
  </data>
 </rpc-reply>
</Body>
</Envelope>
```

XML output element tags

For descriptions of common XML element tags, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

Provisioning element tags in an XML reply are the same as the input element tags. For descriptions, see "XML input element tags" on page 549.

Update requests

Sample XML update request for an E5-110 port (video and data)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1615">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5110</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision>
                  <UserDescr>Frank Smith II</UserDescr>
                  <SubscriberID>555-111-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5110</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL_MAX</DSLProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>1</VideoTemplate>
                  <VideoVLAN>100</VideoVLAN>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>1</DataTemplate>
                  <DataVLAN>103</DataVLAN>
                  <DoubleTagging>0</DoubleTagging>
                </DataProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample XML update request for an E5-111 port (video, data, and voice)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1616">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision>
                  <UserDescr>Frank Smith II</UserDescr>
                  <SubscriberID>555-111-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL_MAX</DSLProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>1</VideoTemplate>
                  <VideoVLAN>100</VideoVLAN>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>1</DataTemplate>
                  <DataVLAN>103</DataVLAN>
                  <DoubleTagging>0</DoubleTagging>
                </DataProvision>
                <VoiceProvision>
                  <VoIPNumber>12345678901</VoIPNumber>
                  <VoiceTemplate>111</VoiceTemplate>
                </VoiceProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input and output element tags for an update request

Important: For update XML requests, a value for each required element tag must be supplied even if the parameter value is not being changed. For example, if data template 2 is in use on a port using data VLAN 103, to update the port to use data template 1, supply both element tags:

```
<DataProvision>
  <DataTemplate>1</DataTemplate>
  <DataVLAN>103</DataVLAN>
</DataProvision>
```

For descriptions of common XML element tags for XML requests, see "Common element tags for E3/E5-100 XML requests" on page 533.

For descriptions of common XML element tags for XML replies, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

The provisioning element tags for an **update** XML request are the same as the element tags in a **create** XML request. For descriptions, see "XML input element tags" on page 549.

Delete requests

Sample XML delete request for an E5-111 port (video, data, and voice)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1620">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
          <request action="delete" userName="rootgod" sessionId="2"
               dependency="false">
            <SSA>
              <NetworkName>NTWK-E5111</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <SSAProvision/>
            </SSA>
```

```
          </request>
          <request action="delete" userName="rootgod" sessionId="2"
               dependency="false">
            <SSA>
              <NetworkName>NTWK-E5111</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>adsl</PortType>
              <VideoProvision/>
              <DataProvision/>
              <VoiceProvision/>
            </SSA>
          </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags for **delete** requests.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char(59)	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>	Int	Yes	Identifies the port number (1 to 48).
<PortType>	see descr	Yes	Identifies the port type. Use the following case-sensitive expression: **adsl**
<SSAProvision/> <VideoProvision/> <DataProvision/> <VoiceProvision/>* * applicable for E5-111 units	N/A	See descr	Include <SSAProvision/> in a separate <request> tag to delete subscriber information from the port. Include one or more of the three service element tags in the left column in <request> tag to delete video, data, or voice provisioning from the port.

Replies to a delete request

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1620">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_111</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

XML output element tags for a **delete** request include <NetworkName>, <PortNumber>, and <PortType>.

E3-12C, E5-120, and E5-121 Service Activation

Sample XML create requests

The following example shows an XML create request with bonding and residential gateway mode disabled, and without double tagging of data traffic. CPE/modem tagging is turned on for the video and data service VLANs.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1621">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                  <DSInp>6</DSInp>
                  <DSPhyR>0</DSPhyR>
                  <USInp>6</USInp>
                  <DSPhyR>0</DSPhyR>
                </PortSetupProvision>
```

```
                <VideoProvision>
                  <VideoTemplate>2</VideoTemplate>
                  <VideoVLAN>5</VideoVLAN>
                  <VideoTxTagging>1</VideoTxTagging>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>2</DataTemplate>
                  <DataVLAN>6</DataVLAN>
                  <DoubleTagging>0</DoubleTagging>
                  <DataTxTagging>1</DataTxTagging>
                </DataProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML create request with bonding and residential gateway mode enabled and with double tagging of data traffic.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1621">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
```

```
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>1</BondingStatus>
                  <ResidentialGateway>1</ResidentialGateway>
                </PortSetupProvision>
                <DataProvision>
                  <DataTemplate>2</DataTemplate>
                  <DataVLAN>6</DataVLAN>
                  <DoubleTagging>1</DoubleTagging>
                  <RCVid>101</RCVid>
                  <Prio>4</Prio>
                  <RCPrio>4</RCPrio>
                </DataProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML create request for an E5-121 port with video, data, and C7 TDM gateway VoIP service. CPE/modem tagging is turned off for the video and data service VLANs.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1622">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
```

```
            <request action="create" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                  <DSInp>6</DSInp>
                  <DSPhyR>0</DSPhyR>
                  <USInp>6</USInp>
                  <DSPhyR>0</DSPhyR>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>2</VideoTemplate>
                  <VideoVLAN>5</VideoVLAN>
                  <VideoTxTagging>0</VideoTxTagging>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>2</DataTemplate>
                  <DataVLAN>6</DataVLAN>
                  <DoubleTagging>0</DoubleTagging>
                  <DataTxTagging>0</DataTxTagging>
                </DataProvision>
                <VoiceProvision>
                  <VoIPNumber>N1-1-IG5-33</VoIPNumber>
                  <VoiceTemplate>1</VoiceTemplate>
                  <CSIPNetworkName>NTWK-DMZ</CSIPNetworkName>
                  <CSIPVCG>N1-1-IG2</CSIPVCG>
                </VoiceProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML create request with VLAN translation of video and data traffic.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1622">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="E5100">
```

```
            <request action="create" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="create" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>80</VideoTemplate>
                  <VideoVLAN>1005</VideoVLAN>
                  <CVid>4</CVID>
                  <PBit>5</PBit>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>80</DataTemplate>
                  <DataVLAN>0</STagBit>
                  <STagBit>3</STagPBit>
                  <CVid>24</CVid>
                  <RCVid>4</RCVID>
                  <PBit>5</PBit>
                </DataProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags: E3-12C, E5-120 and E5-121 service units

The following table lists the XML element tags for **create** requests for E3-12C, E5-120, and E5-121 service units. For element tags with E3-48 service units, see "Replies to an create request" on page 568.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>* * For bonding groups, use the master port. For rules on master port designation, refer to the E3/E5-100 user documentation.	Int	Yes	Identifies the port number: • E3-12C: 1 to 12. • E3-120 and E5-121: 1 to 24.
<PortType>	see descr	Yes	Identifies the port type. Use the following case-sensitive expression: **vdsl**
<SSAProvision> <UserDescr> <SubscriberID>	see descr	See descr	Following company policy and procedures, include these tags in a separate <request> tag to add a description and subscriber ID to the record: <UserDescr> is a Char(31) user-defined field for subscriber information, such as a name. Spaces and dashes are permitted. <SubscriberID> is Char(15) user-defined field for the subscriber ID, such as a telephone number. Spaces and dashes are permitted.

Element Tag	Data Type	Req'd ?	Description
<PortSetupProvision> <DSLMode> <DSLProfile> <IPQoSProfile> <BondingStatus> <ResidentialGateway>* <DSInp> <DSPhyR> <USInp> <DSPhyR> * When residential gateway mode is enabled, the <VideoProvision> tags are not supplied; for this mode, the template used for the <DataTemplate> tag must include the multicast VLAN (MVLAN).	 Int Char Char Bool Bool Int Bool Int Bool	 Yes Yes Yes Yes Yes No No No No	<DSLMode> identifies the VDSL mode using one of the following: • **1**—VDSL Profile 8a • **2**—VDSL Profile 8b • **3**—VDSL Profile 8c • **4**—VDSL Profile 8d • **5**—VDSL Profile 12a • **6**—VDSL Profile 12b • **7**—VDSL Profile 17a • **8**—Auto • **9**—ADSL2+ • **10**—VDSL2 • **11**—G.DMT • **12**—G.Lite • **13**—ADSL2 • **14**—T1.413 • **15**—VDSL2 + fallback to ADSL2 • **16**—VDSL2 + fallback to G.DMT • **17**—VDSL2 + fallback to G.Lite • **18**—VDSL2 + fallback to T1.413 <DSLProfile> identifies the xDSL profile name.** <IPQoSProfile> identifies the IP QoS profile name.** <BondingStatus> identifies whether bonding is enabled (1) or disabled (0) for the port. See the note above under <PortNumber> regarding master ports. <ResidentialGateway> identifies whether residential gateway mode is enabled (1) or disabled (0).* <DSInp> and <USInp> identify the downstream and upstream impulse noise protection values (5 to 160; 0 to disable). If not supplied, the default (5) is used. <DSPhyR> and <USPhyR> identify whether PhyR is enabled (1) or disabled (0). If not supplied, 0 is used. ** Up to 31 characters, excluding spaces and dashes. Profile names prefixed by the @ symbol designate global CMS profiles.

Element Tag	Data Type	Req'd ?	Description
<VideoProvision> <VideoTemplate> <VideoVLAN> <VideoTxTagging> <CVid>* <PBit>* * Required when VLAN tag actions are applied. **Note:** When residential gateway mode is enabled, do not include the <VideoProvision> tags.	 Int Int Bool Int Char	Yes	<VideoTemplate> identifies the video subscriber template (1 to 500). <VideoVLAN> identifies the video service VLAN (1 to 4094). <VideoTxTagging> identifies whether transmit tagging for the video VLAN is turned on or off: • **0**—off: VLAN tags are not required on the CPE/modem for video service • **1**—on: VLAN tags are required on the CPE/modem for video service When tag actions are applied on the video service VLAN, supply these two tags: <CVid> identifies the customer service VLAN at the CPE (1 to 4094). <PBit> identifies the video service VLAN priority bit to match: • 0 to 7—specify a priority • **any—**do not match priority
Data provisioning when double tagging and VLAN translation are not applied: <DataProvision> <DataTemplate> <DataVLAN> <DoubleTagging> <DataTxTagging>	 Int Int Bool Bool	Yes	<DataTemplate> identifies the data subscriber template (1 to 500). <DataVLAN> identifies the data service VLAN (1 to 4094). <DoubleTagging> identifies whether double tagging is turned on or off: • **0**—off <DataTxTagging> identifies whether transmit tagging for the data VLAN is turned on or off: • **0**—off: VLAN tags are not required on the CPE/modem for data service • **1**—on: VLAN tags are required on the CPE/modem for data service

Element Tag	Data Type	Req'd ?	Description

Data provisioning when double tagging is enabled and no VLAN translation is applied:
 	
`<DataProvision>`
`  <DataTemplate>`
`  <DataVLAN>`
`  <DoubleTagging>`
`  <RCVid>`
`  <Prio>`
`  <RCPrio>`
	
	`Int`
	`Bool`
	`Bool`
	`Int`
	`Int`
	`Int`
		Yes	`<DataTemplate>` identifies the data subscriber template (1 to 500).
			`<DataVLAN>` identifies the data service VLAN (1 to 4094).
			`<DoubleTagging>` identifies whether double tagging is turned on or off:
			• **1**—on
			`<RCVid>`: CVID (1 to 4094)
			`<Prio>`: SVID Priority bit (0 to 7)
			`<RCPrio>`: CVID Priority bit (0 to 7)

Data provisioning when VLAN translation is applied:
 	
`<DataProvision>`
`  <DataTemplate>`
`  <DataVLAN>`
`  <DoubleTagging>`
`  <CVid>`
`  <RCVid>`
	
	`Int`
	`Int`
	`Bool`
	`Int`
	`Int`
		Yes	`<DataTemplate>` identifies the data subscriber template (1 to 500).
			`<DataVLAN>` identifies the data service VLAN or SVID (1 to 4094).
			`<DoubleTagging>` identifies whether double tagging is turned on or off:
			• **0**—off (must be off when tag actions are applied)
			`<CVid>`: CXVID, or the VLAN you are translating into (1 to 4094)
			`<RCVid>`: CVID (1 to 4094)

E5-121 or E3-12C SIP VoIP service:
`<VoiceProvision>`
`  <VoIPNumber>`
`  <VoiceTemplate>`
`  <VoIPUserName>*`
`  <VoIP Password>*`
 	
* When <VoIPUserName> and <VoIPPassword> are supplied, a new local call services profile is created using the attributes of the global Call Services profile specified in the voice subscriber template. The <VoIPUserName> value is used as the new call services profile name as well as the port SIP authorization user name.
	
	`Char`
	`Int`
	`Char`
	`Char`
		For SIP voice service (see descr)	`<VoIPNumber>` identifies the user ID (up to 15 characters). The value can be the SIP user name (for URI type = SIP) or telephone number (for URI type = Tel).
			`<VoiceTemplate>` identifies the voice subscriber template (1 to 500).
			`<VoIPUserName>` optionally identifies the user name to use for authentication. See the note to the left.
			`<VoIPPassword>` optionally identifies the password to use for authentication. See the note to the left.

Element Tag	Data Type	Req'd ?	Description
E5-121 or E3-12C C7 TDM Gateway VoIP service: <VoiceProvision> <VoIPNumber> <VoiceTemplate> <CSIPNetworkName> <CSIPVCG>	 Char Int Char Char	For C7 TDM Gateway voice service	<VoIPNumber> identifies the case-sensitive CRV or channel ID for the subscriber line, as provisioned on the C7 GR-303 or GR-8 interface group. Example: N1-1-IG1-224 **Tip:** Use a voice template that has a pre-defined TDM service group to prevent duplicate CRVs across service units. <VoiceTemplate> identifies the voice subscriber template (1 to 500). <CSIPNetworkName> identifies the case-sensitive name of the C7 network providing TDM gateway service, preceded by NTWK-. Example: NTWK-Pet01_C7 <CSIPVCG> identifies the case-sensitive AID of the C7 SIP voice concentration group. Example: N1-1-IG2
E5-121 or E3-12C H.248 VoIP service: <VoiceProvision> <VoiceTemplate>	 Int	For H.248 voice service	<VoiceTemplate> identifies the voice subscriber template (1 to 500).

Replies to an create request

Sample XML create reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1623">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>adsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
```

```
            </SSA>
          </resultList>
        </response>
        <response>
          <ResultCode>0</ResultCode>
          <ResultMessage/>
          <resultList>
            <SSA>
              <NetworkName>NTWK-E5_121</NetworkName>
              <PortNumber>1</PortNumber>
              <PortType>vdsl</PortType>
              <PortSetupProvision>
                <DSLMode>4</DSLMode>
                <DSLProfile>DEFVAL</DSLProfile>
                <IPQosProfile>DEFVAL</IPQosProfile>
                <BondingStatus>0</BondingStatus>
                <ResidentialGateway>0</ResidentialGateway>
                <DSInp>6</DSInp>
                <DSPhyR>0</DSPhyR>
                <USInp>6</USInp>
                <DSPhyR>0</DSPhyR>
                <PortSetupActive>1</PortSetupActive>
                <PortSetupErrMsg/>
              </PortSetupProvision>
              <VideoProvision>
                <VideoTemplate>2</VideoTemplate>
                <VideoVLAN>5</VideoVLAN>
                <VideoActive>1</VideoActive>
              </VideoProvision>
              <DataProvision>
                <DataTemplate>2</DataTemplate>
                <DataVLAN>6</DataVLAN>
                <DoubleTagging>0</DoubleTagging>
                <DataActive>1</DataActive>
              </DataProvision>
              <VoiceProvision>
                <VoIPNumber>98765432</VoIPNumber>
                <VoiceTemplate>1</VoiceTemplate>
                <VoiceActive>1</VoiceActive>
              </VoiceProvision>
            </SSA>
          </resultList>
        </response>
      </responses>
    </data>
  </rpc-reply>
 </Body>
</Envelope>
```

XML output element tags

For descriptions of common XML element tags, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

Provisioning element tags in an XML reply are the same as the input element tags. For descriptions, see "XML input element tags: E3-12C, E5-120 and E5-121 service units" on page 564.

Update requests

Sample XML update request for an E5-120 port (video and data)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1624">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="615">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith Esq</UserDescr>
                  <SubscriberID>555-111-4444</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="update" userName="rootgod" sessionId="615">
              <SSA>
                <NetworkName>NTWK-E5_120</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL_MAX</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>1</VideoTemplate>
                  <VideoVLAN>100</VideoVLAN>
                </VideoProvision>
                <DataProvision>
```

```
                <DataTemplate>1</DataTemplate>
                <DataVLAN>103</DataVLAN>
                <DoubleTagging>0</DoubleTagging>
              </DataProvision>
            </SSA>
          </request>
        </requests>
      </netconf:config>
    </netconf:edit-config>
  </netconf:rpc>
 </soapenv:Body>
</soapenv:Envelope>
```

Sample XML update request for an E5-121 port (video, data, and voice)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1625">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith Esq</UserDescr>
                  <SubscriberID>555-111-4444</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="615" >
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL_MAX</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>1</VideoTemplate>
                  <VideoVLAN>100</VideoVLAN>
                </VideoProvision>
```

```
                <DataProvision>
                  <DataTemplate>1</DataTemplate>
                  <DataVLAN>103</DataVLAN>
                  <DoubleTagging>0</DoubleTagging>
                </DataProvision>
                <VoiceProvision>
                  <VoIPNumber>12345678901</VoIPNumber>
                  <VoiceTemplate>1</VoiceTemplate>
                </VoiceProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input and output element tags for an update request

Important: For update XML requests, a value for each required element tag must be supplied even if the parameter value is not being changed. For example, if data template 2 is in use on a port using data VLAN 103, to update the port to use data template 1, supply both element tags:

```
<DataProvision>
  <DataTemplate>1</DataTemplate>
  <DataVLAN>103</DataVLAN>
</DataProvision>
```

For descriptions of common XML element tags for XML requests, see "Common element tags for E3/E5-100 XML requests" on page 533.

For descriptions of common XML element tags for XML replies, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

The provisioning element tags for an **update** XML request are the same as the element tags in a **create** XML request. For descriptions, see "XML input element tags: E3-12C, E5-120 and E5-121 service units" on page 564.

XML delete requests

Sample XML delete request for an E5-121 port (video, data, and voice)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
        message-id="1626">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="delete" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <SSAProvision/>
              </SSA>
            </request>
          <requests module="E5100">
            <request action="delete" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
                <VoiceProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags for **delete** requests.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>	Int	Yes	Identifies the port number. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortType>	see descr	Yes	Identifies the port type. Use the following case-sensitive expression: **vdsl**
<SSAProvision/> <VideoProvision/> <DataProvision/> <VoiceProvision/>* * applicable for E3-12C and E5-121 units	N/A		Include <SSAProvision/> in a separate <request> tag to delete subscriber information from the port. Include one or more of the three service element tags in the left column in <request> tag to delete video, data, or voice provisioning from the port.

Replies to a delete request

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1626">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

XML output element tags for a **delete** request include <NetworkName>, <PortNumber>, and <PortType>.

E3-48 Service Activation

Sample XML create requests

The following example shows an XML create request with bonding and residential gateway mode disabled, and with double tagging of data traffic.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1623">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E3_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E3_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <CoSQProfile>DEFVAL</CoSQProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingPortNumber>18</BondingPortNumber>
                  <BondingGroupName>GRP8</BondingGroupName>
                  <ResidentialGateway>0</ResidentialGateway>
                  <DSInp>6</DSInp>
                  <DSPhyR>0</DSPhyR>
                  <USInp>6</USInp>
                  <DSPhyR>0</DSPhyR>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>2</VideoTemplate>
                  <VideoVLAN>5</VideoVLAN>
                  <VideoTxTagging>1</VideoTxTagging>
```

```
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>2</DataTemplate>
                  <DataVLAN>6</DataVLAN>
                  <DoubleTagging>1</DoubleTagging>
                  <RCVid>3</RCVid>
                  <Prio>3</Prio>
                  <RCPrio>3</RCPrio>
                </DataProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

The following example shows an XML create request for a bonding group, with service tag actions for video and data traffic.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1623">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E3_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith</UserDescr>
                  <SubscriberID>666-222-3333</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="create" userName="rootgod" sessionId="2"
                 dependency="true">
              <SSA>
                <NetworkName>NTWK-E3_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>8</DSLMode>
                  <DSLProfile>DEFVAL</DSLProfile>
                  <CoSQProfile>DEFVAL</CoSQProfile>
```

```
                    <IPQosProfile>DEFVAL</IPQosProfile>
                    <BondingPortNumber>18</BondingPortNumber>
                    <BondingGroupName>GRP8</BondingGroupName>
                    <ResidentialGateway>0</ResidentialGateway>
                  </PortSetupProvision>
                  <VideoProvision>
                    <VideoTemplate>80</VideoTemplate>
                    <VideoVLAN>1005</VideoVLAN>
                    <CVid>4</CVID>
                    <PBit>5</PBit>
                  </VideoProvision>
                  <DataProvision>
                    <DataTemplate>80</DataTemplate>
                    <DataVLAN>0</STagBit>
                    <STagBit>3</STagPBit>
                    <CVid>24</CVid>
                    <RCVid>4</RCVID>
                    <PBit>5</PBit>
                  </DataProvision>
                </SSA>
              </request>
            </requests>
          </netconf:config>
        </netconf:edit-config>
      </netconf:rpc>
    </soapenv:Body>
  </soapenv:Envelope>
```

XML input element tags: E3-48 service units

The following table lists the XML element tags for **create** requests.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>* * For bonding groups, use the master port. For rules on master port designation, refer to the Calix E3-48 user documentation.	Int	Yes	Identifies the port number (1 to 48).
<PortType>	see descr	Yes	Identifies the port type. Use the following case-sensitive expression: **vdsl**
<SSAProvision> <UserDescr> <SubscriberID>	See descr	See descr	Following company policy and procedures, include these tags in a separate <request> tag to add a description and subscriber ID to the record: <UserDescr> is a Char(31) user-defined field for subscriber information, such as a name. Spaces and dashes are permitted. <SubscriberID> is a Char(15) user-defined field for the subscriber ID, such as a telephone number. Spaces and dashes are permitted.

Element Tag	Data Type	Req'd ?	Description
<PortSetupProvision> <DSLMode> <DSLProfile> <CoSQProfile> <IPQoSProfile> <BondingPortNumber> <BondingGroupName> <ResidentialGateway>* <DSInp> <DSPhyR> <USInp> <DSPhyR> **Notes:** * When residential gateway mode is enabled, the <VideoProvision> tags are not supplied; for this mode, the template used for the <DataTemplate> tag must include the multicast VLAN (MVLAN).	 Int Char Char Char Int Char Bool Int Bool Int Bool	 Yes Yes Yes Yes Yes Yes Yes No No No No	<DSLMode> identifies the VDSL mode using one of the following: • **1**—VDSL2 Profile 8a • **2**—VDSL2 Profile 8b • **3**—VDSL2 Profile 8c • **4**—VDSL2 Profile 8d • **5**—VDSL2 Profile 12a • **6**—VDSL2 Profile 12b • **7**—VDSL2 Profile 17a • **8**—Auto • **9**—Auto ADSL2+ Fallback • **10**—VDSL2 • **19**—VDSL2 8a+ Fallback to ADSL2+ <DSLProfile> identifies the xDSL profile name.** <CoSQProfile> identifies the CoS queue profile name.** <DSLProfile> identifies the xDSL profile name (up to 31 characters, excluding spaces and dashes).** <BondingPortNumber> identifies the slave port in the bonding group. <BondingGroupName> identifies the bonding group name (up to 31 characters, excluding spaces). <ResidentialGateway> identifies whether residential gateway mode is enabled (1) or disabled (0) for the port (see note to left). <DSInp> and <USInp> identify the downstream and upstream impulse noise protection (INP) values (5 to 160; 0 to disable). If not supplied, the default (5) is provisioned on the port. <DSPhyR> and <USPhyR> identify whether PhyR is enabled (1) or disabled (0) for the port. If not supplied, 0 is provisioned on the port. ** Up to 31 characters, excluding spaces and dashes. Profile names prefixed by the @ symbol designate global CMS profiles.

Element Tag	Data Type	Req'd ?	Description
<VideoProvision> <VideoTemplate> <VideoVLAN> <VideoTxTagging> <CVid>* <PBit>* * Required when VLAN tag actions are applied. **Note:** When residential gateway mode is enabled, do not include the <VideoProvision> tags.	 Int Int Bool Int Char	Yes	<VideoTemplate> identifies the video subscriber template (1 to 500). <VideoVLAN> identifies the video service VLAN (1 to 4094). <VideoTxTagging> identifies whether transmit tagging for the video VLAN is turned on or off: • **0**—off: VLAN tags are not required on the CPE/modem for video service • **1**—on: VLAN tags are required on the CPE/modem for video service When tag actions are applied on the video service VLAN, supply these two tags: <CVid> identifies the video service VLAN at the CPE (1 to 4094). <PBit> identifies the video service VLAN priority bit to match: • 0 to 7—specify a priority • **any—**do not match priority
Data provisioning when double tagging and tag actions are not applied: <DataProvision> <DataTemplate> <DataVLAN> <DoubleTagging> <DataTxTagging>	 Int Int Bool Bool	Yes	<DataTemplate> identifies the data subscriber template (1 to 500). <DataVLAN> identifies the data service VLAN (1 to 4094). <DoubleTagging> identifies whether double tagging is turned on or off: • **0**—off <DataTxTagging> identifies whether transmit tagging for the data VLAN is turned on or off: • **0**—off: VLAN tags are not required on the CPE/modem for data service • **1**—on: VLAN tags are required on the CPE/modem for data service

Element Tag	Data Type	Req'd ?	Description
Data provisioning when double tagging is enabled and no tag actions are applied: <DataProvision> <DataTemplate> <DataVLAN> <RCVid> <Prio> <RCPrio>	 Int Int Int Int Int Int	Yes	<DataTemplate> identifies the data subscriber template (1 to 500). **<DataVLAN>** identifies the data service VLAN (1 to 4094). **<RCVid>**: CVID (1 to 4094) **<Prio>**: SVID Priority bit (0 to 7) **<RCPrio>**: CVID Priority bit (0 to 7)
Data provisioning when tag actions are applied: <DataProvision> <DataTemplate> <DataVLAN> <DoubleTagging> <STagPBit> <CVid> <RCVid> <PBit>	 Int Int Bool Int Int Int Int	Yes	<DataTemplate> identifies the data subscriber template (1 to 500). **<DataVLAN>** identifies the data service VLAN or SVID (1 to 4094). **<DoubleTagging>** identifies whether double tagging is turned on or off: • **0**—off (must be off when tag actions are applied) **<STagPBit>** Data service tag priority bit (0 to 7) **<CVid>**: CXVID, or the VLAN you are translating from (1 to 4094) **<RCVid>**: CVID (1 to 4094) **<PBit>**: Matching priority bit specified on the data subscriber template: • 0 to 7—specify a priority • **any—**do not match priority

Replies to an create request

Sample XML create reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/><Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1623">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
```

```
            <NetworkName>NTWK-E5_111</NetworkName>
            <PortNumber>1</PortNumber>
            <PortType>adsl</PortType>
            <SSAProvision>
              <UserDescr>Sally Smith</UserDescr>
              <SubscriberID>666-222-3333</SubscriberID>
            </SSAProvision>
          </SSA>
        </resultList>
      </response>
      <response>
        <ResultCode>0</ResultCode>
        <ResultMessage/>
        <resultList>
          <SSA>
            <NetworkName>NTWK-E5_121</NetworkName>
            <PortNumber>1</PortNumber>
            <PortType>vdsl</PortType>
            <PortSetupProvision>
              <DSLMode>4</DSLMode>
              <DSLProfile>DEFVAL</DSLProfile>
              <IPQosProfile>DEFVAL</IPQosProfile>
              <BondingStatus>0</BondingStatus>
              <ResidentialGateway>0</ResidentialGateway>
              <DSInp>6</DSInp>
              <DSPhyR>0</DSPhyR>
              <USInp>6</USInp>
              <DSPhyR>0</DSPhyR>
              <PortSetupActive>1</PortSetupActive>
              <PortSetupErrMsg/>
            </PortSetupProvision>
            <VideoProvision>
              <VideoTemplate>2</VideoTemplate>
              <VideoVLAN>5</VideoVLAN>
              <VideoActive>1</VideoActive>
            </VideoProvision>
            <DataProvision>
              <DataTemplate>2</DataTemplate>
              <DataVLAN>6</DataVLAN>
              <DoubleTagging>0</DoubleTagging>
              <DataActive>1</DataActive>
            </DataProvision>
            <VoiceProvision>
              <VoIPNumber>98765432</VoIPNumber>
              <VoiceTemplate>1</VoiceTemplate>
              <VoiceActive>1</VoiceActive>
            </VoiceProvision>
          </SSA>
        </resultList>
      </response>
    </responses>
  </data>
 </rpc-reply>
</Body>
</Envelope>
```

XML output element tags

For descriptions of common XML element tags, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

Provisioning element tags in an XML reply are the same as the input element tags. For descriptions, see "XML input element tags: E3-48 service units" on page 579.

Update requests

Sample XML update request for an E3-48 port (video and data)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1624">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="615">
              <SSA>
                <NetworkName>NTWK-E5_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr>Sally Smith Esq</UserDescr>
                  <SubscriberID>555-111-4444</SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
            <request action="update" userName="rootgod" sessionId="615">
              <SSA>
                <NetworkName>NTWK-E3_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision>
                  <DSLMode>4</DSLMode>
                  <DSLProfile>DEFVAL_MAX</DSLProfile>
                  <IPQosProfile>DEFVAL</IPQosProfile>
                  <BondingStatus>0</BondingStatus>
                  <ResidentialGateway>0</ResidentialGateway>
                </PortSetupProvision>
                <VideoProvision>
                  <VideoTemplate>1</VideoTemplate>
                  <VideoVLAN>100</VideoVLAN>
                </VideoProvision>
                <DataProvision>
                  <DataTemplate>1</DataTemplate>
```

```
                <DataVLAN>103</DataVLAN>
                <DoubleTagging>0</DoubleTagging>
              </DataProvision>
            </SSA>
          </request>
        </requests>
      </netconf:config>
    </netconf:edit-config>
  </netconf:rpc>
</soapenv:Body>
</soapenv:Envelope>
```

XML input and output element tags for an update request

Important: For update XML requests, a value for each required element tag must be supplied even if the parameter value is not being changed. For example, if data template 2 is in use on a port using data VLAN 103, to update the port to use data template 1, supply both element tags:

```
<DataProvision>
  <DataTemplate>1</DataTemplate>
  <DataVLAN>103</DataVLAN>
</DataProvision>
```

For descriptions of common XML element tags for XML requests, see "Common element tags for E3/E5-100 XML requests" on page 533.

For descriptions of common XML element tags for XML replies, including the status element tag nested in each of the four provisioning element tags, see "Common XML element tags for E3/E5-100 replies" on page 537.

The provisioning element tags for an **update** XML request are the same as the element tags in a **create** XML request. For descriptions, see "XML input element tags: E3-48 service units" on page 579.

XML delete requests

Sample XML delete request for an E3-48 port (video, data, and voice)

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1626">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url/>
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="delete" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <SSAProvision/>
              </SSA>
            </request>
          <requests module="E5100">
            <request action="delete" userName="rootgod" sessionId="2">
              <SSA>
                <NetworkName>NTWK-E5_48</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
                <PortSetupProvision/>
                <VideoProvision/>
                <DataProvision/>
                <VoiceProvision/>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

XML input element tags

The following table lists the XML element tags for **delete** requests.

Element Tag	Data Type	Req'd ?	Description
<NetworkName>	Char	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortNumber>	Int	Yes	Identifies the port number. See "Common element tags for E3/E5-100 XML requests" on page 533.
<PortType>	see descr	Yes	Identifies the port type. Use the following case-sensitive expression: **vdsl**
<SSAProvision/> <VideoProvision/> <DataProvision/>	N/A		Include <SSAProvision/> in a separate <request> tag to delete subscriber information from the port. Include one or more of the service element tags in the left column in <request> tag to delete video or data provisioning from the port.

Replies to a delete request

Sample XML delete reply

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header/>
  <Body>
    <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1626">
      <data>
        <responses xmlns="http://schema.calx.com/nc/msoe">
          <response>
            <ResultCode>0</ResultCode>
            <ResultMessage/>
            <resultList>
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>1</PortNumber>
                <PortType>vdsl</PortType>
              </SSA>
            </resultList>
          </response>
        </responses>
      </data>
    </rpc-reply>
  </Body>
</Envelope>
```

XML output element tags

XML output element tags for a **delete** request include <NetworkName>, <PortNumber>, and <PortType>.

Suspending and Resuming Service

This section defines the objects required to suspend and resume video and data service on xDSL ports or voice service on VoIP ports. A successful suspend request disables all services provisioned on the xDSL or VoIP port, and a successful resume requests enables the provisioned services. For a bonding group, the port identified in the requests must be the master port.

Suspend and resume requests are sent as **update** requests with an additional actionName attribute in the <request> tag and an <SSA> tag nested under the <request> tab. For more information see "Common element tags for E3/E5-100 XML requests" on page 533, and the examples below.

Sample XML suspend and resume service requests

The following is an example of an **update** request to suspend all services on an xDSL port 5:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1627">
      <netconf:edit-config>
        <netconf:target><netconf:url /></netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="1001"
                 actionName="suspend">
              <SSA>
                <ID>5</ID>
                <NetworkName>NTWK-E3_48</NetworkName>
                <ServiceType>video-data</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

The following is an example of an **update** request to resume voice service on a VoIP port 5:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        message-id="1628">
      <netconf:edit-config>
        <netconf:target><netconf:url/></netconf:target>
        <netconf:config>
          <requests module="Msap">
            <request action="update" userName="rootgod" sessionId="1001"
                 actionName="resume">
              <SSA>
                <ID>5</ID>
                <NetworkName>NTWK-E5_121</NetworkName>
                <ServiceType>voice</ServiceType>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
<soapenv:Envelope>
```

XML input element tags and values

The following tables list the XML element tags in a **update** request for suspending and resuming service.

Element Tags	Data Type	Req'd ?	Description
<ID>	Char	Yes	Identifies the xDSL or VoIP port number. • E3-48 and E5-110/111: 1 to 48 • E5-120/121: 1 to 24 • E3-12C: 1 to 12 **Note:** For a bonding group, use the port number of the master port.
<NetworkName>	Char(59)	Yes	Identifies the service unit. See "Common element tags for E3/E5-100 XML requests" on page 533.
<ServiceType>	Int	Yes	Identifies the type of service to be suspended or resumed using one of the following case-sensitive expressions: • **video-data** • **voice**

Removing Subscriber Information

This section shows the objects required to remove subscriber information. As shown below, leave the UserDescr and SubscriberID fields blank to remove this subscriber information from a port interface.

Sample XML suspend and resume service requests

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <netconf:rpc xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" message-id="3">
      <netconf:edit-config>
        <netconf:target>
          <netconf:url />
        </netconf:target>
        <netconf:config>
          <requests module="E5100">
            <request action="update" userName="rootgod" sessionId="52">
              <SSA>
                <NetworkName>NTWK-E5_121</NetworkName>
                <PortNumber>23</PortNumber>
                <PortType>vdsl</PortType>
                <SSAProvision>
                  <UserDescr></UserDescr>
                  <SubscriberID></SubscriberID>
                </SSAProvision>
              </SSA>
            </request>
          </requests>
        </netconf:config>
      </netconf:edit-config>
    </netconf:rpc>
  </soapenv:Body>
</soapenv:Envelope>
```

PostgreSQL to REST/JSON API

This chapter provides information about the solution to execute SQL to CMS using postgREST and covers the following topics:

Guidelines for PostgREST API Provisioning

When using the examples and explanations in this guide, keep in mind the following:

- **Local system installation requirement** CentOS 6, 64 bit
  Note: If your CMS system is running CentOS 6 64 bit, you can deploy this tool to the same server as your CMS system.

Setting Up the PostgREST Tool

This topic provides instruction for how to set up this tool.

1. Download the postgrest.tar.gz attachment:

http://wiki.calix.local/pages/viewpage.action?pageId=55642500&preview=/55642500/56303275/postgrest.tar.gz

2. Extract the postgrest.tar.gz file to the desired directory.

Recommended directory: /opt/PostgREST

tar -xzvf postgrest.tar.gz

Using the PostgREST Tool (Local Solution)

To run the PostgREST tool

1. Run the postgREST that you extracted:

```
./postgrest
```

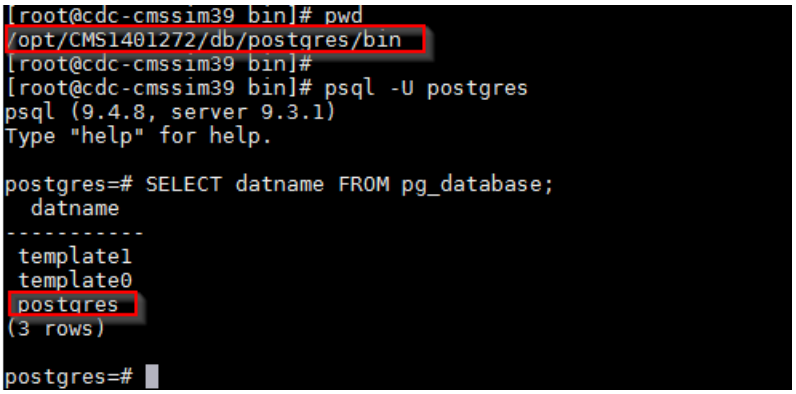
When the result appears as shown in the example below, the compiled binary will work as expected.

```
[root@cdc-cmssim39 bin]# ./postgrest
Missing: DB_URL (-a|--anonymous ROLE)

Usage: postgrest DB_URL (-a|--anonymous ROLE) [-x|--proxy-uri PROXY]
                 [-s|--schema NAME] [-l|--host HOST] [-p|--port PORT]
                 [-j|--jwt-secret SECRET] [-o|--pool COUNT]
                 [-m|--max-rows COUNT]
  PostgREST 0.3.2.0 / create a REST API to an existing Postgres database

Available options:
  -h,--help                Show this help text
  DB_URL                   (REQUIRED) database connection string, e.g.
                           postgres://user:pass@host:port/db
  -a,--anonymous ROLE      (REQUIRED) postgres role to use for non-authenticated
                           requests
  -x,--proxy-uri PROXY     proxy uri of the HTTP server
  -s,--schema NAME         schema to use for API routes (default: "public")
  -l,--host HOST           hostname or ip on which to run HTTP
                           server (default: "*4")
  -p,--port PORT           port number on which to run HTTP
                           server (default: 3000)
  -j,--jwt-secret SECRET   secret used to encrypt and decrypt JWT
                           tokens (default: "secret")
  -o,--pool COUNT          max connections in database pool (default: 10)
  -m,--max-rows COUNT      max rows in response (default: "infinity")
[root@cdc-cmssim39 bin]#
```

To deploy the postgREST tool

1. Create a read-only database user on the CMS postgREST database.

   a. **Path:** CMS_INSTALL_PATH/db/postgres/bin



   b. **Command:** CMS_INSTALL_PATH/bin/createdbuser.sh readonly_user readonly
   Note: When CMS drops, creates, or alters a table, the readonly_user may lose permission to access those tables. Solution:
   create a cron job to execute [grantdbuser.sh readonly_user] every 5 minutes

```
[root@cdtlnx01 log]#cd /var/spool/cron/
[root@cdtlnx01 cron]#ls
cmsuser root
[root@cdtlnx01 cron]#vim cmsuser

*/5 * * * * sh CMS_INSTALL_PATH/bin/grantdbuser.sh readonly_user
```

2. Start up PostgREST

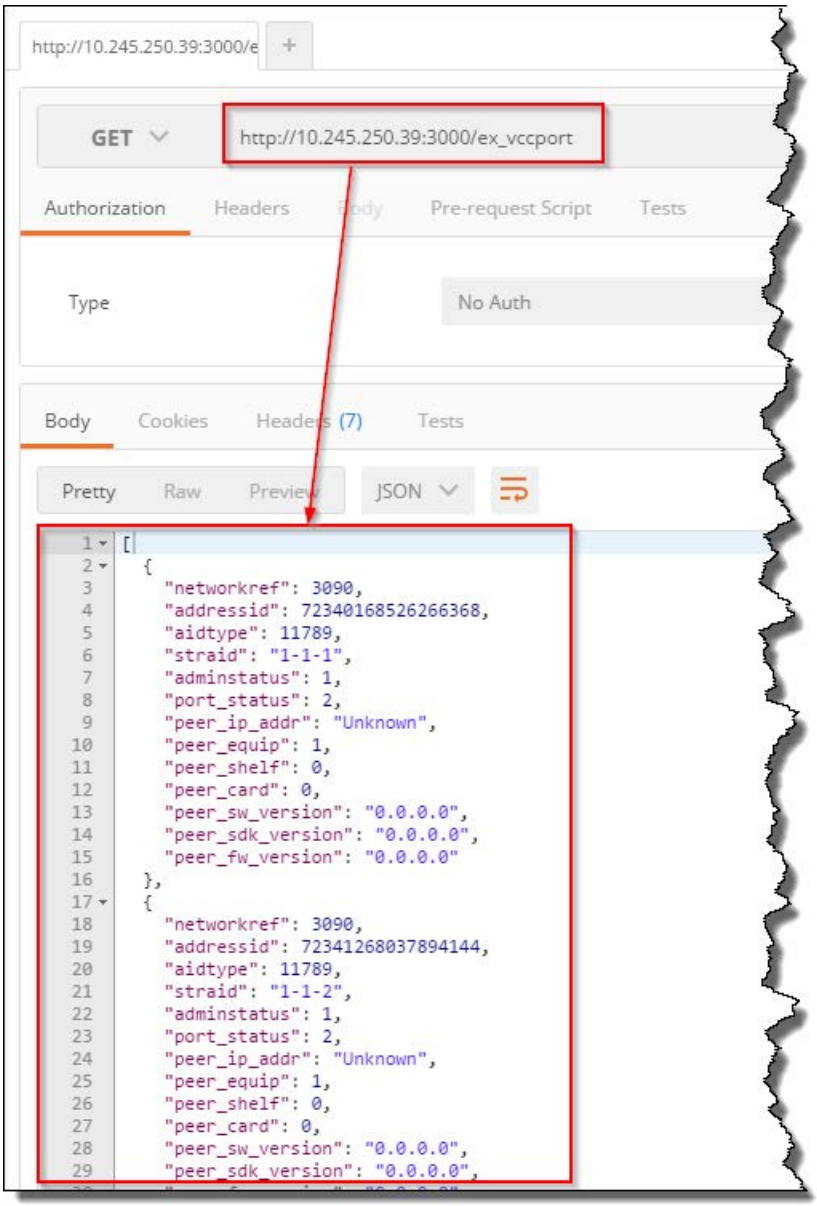
```
./postgrest "user=readonly_user password=readonly host=cms_server_ip
port=5432 dbname=postgres" -a readonly_user --schema=public
```

   Note: Because PostgREST and CMS are on the same server, you can use the cms_server_ip as the localhost.

3. Query data via PostgREST

```
GET http://postgrest_server_ip:3000/table_name
```

Note: table_name must be all lowercase.



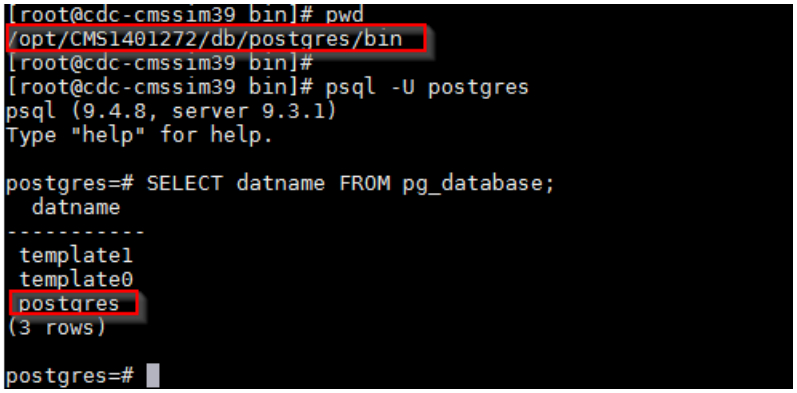
Using the PostgREST Tool (Remote Solution)

A remote solution is when the PostgREST is deployed on an independent server, not on the server where CMS is deployed.

To deploy the PostgREST tool

1. Create a read only database user on the CMS postgREST database.

   a. **Path:** CMS_INSTALL_PATH/db/postgres/bin



   b. **Command:** CMS_INSTALL_PATH/bin/createdbuser.sh readonly_user readonly
   Note: When CMS drops, creates, or alters a table, the readonly_user may lose permission to access those tables. Solution:
   create a cron job to execute [grantdbuser.sh readonly_user] every 5 minutes.

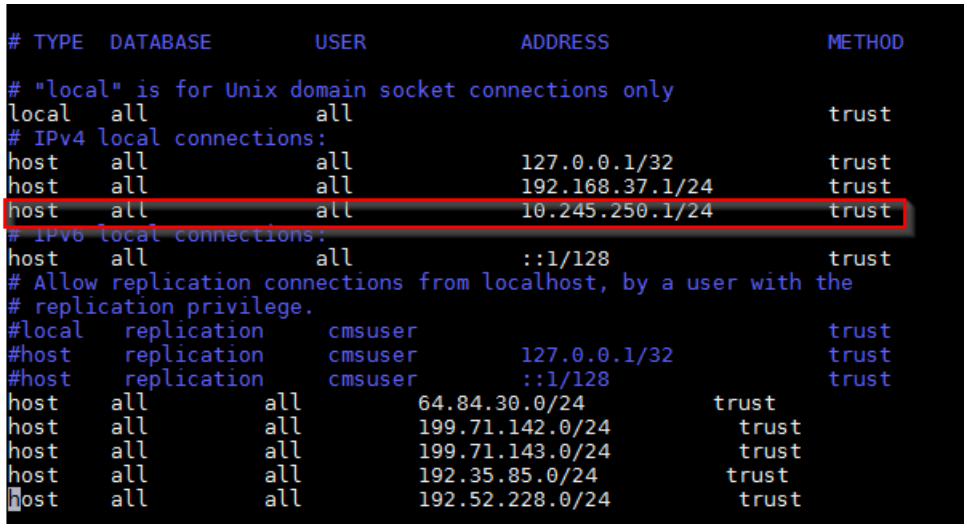
```
[root@cdtlnx01 log]#cd /var/spool/cron/
[root@cdtlnx01 cron]#ls
cmsuser root
[root@cdtlnx01 cron]#vim cmsuser

*/5 * * * * sh CMS_INSTALL_PATH/bin/grantdbuser.sh readonly_user
```

2. Enable CMS postgres tcp/ip connection.

   a. **Path:** CMS_INSTALL_PATH/db/postgres/data/pg_hba.conf

   b. Add below record under #IPv4 local connections, to allow PostgREST visit the CMS database.

```
host    all         all         postgrest_server_ip/mask          trust
```



```
# TYPE  DATABASE        USER            ADDRESS                 METHOD

# "local" is for Unix domain socket connections only
local   all             all                                     trust
# IPv4 local connections:
host    all             all             127.0.0.1/32            trust
host    all             all             192.168.37.1/24         trust
host    all             all             10.245.250.1/24         trust
# IPv6 local connections:
host    all             all             ::1/128                 trust
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local   replication     cmsuser                                 trust
#host    replication     cmsuser         127.0.0.1/32            trust
#host    replication     cmsuser         ::1/128                 trust
host    all         all         64.84.30.0/24          trust
host    all         all         199.71.142.0/24          trust
host    all         all         199.71.143.0/24          trust
host    all         all         192.35.85.0/24         trust
host    all         all         192.52.228.0/24          trust
```

c. These actions take effect when the database is shut down and restarted:

- To stop CMS postgres:

  Path: CMS_INSTALL_PATH/bin/stoppostgres.sh

- To startup CMS postgres:

  Path: CMS_INSTALL_PATH/bin/startpostgres.sh

3. Startup PostgREST:

```
./postgrest "user=readonly_user password=readonly host=cms_server_ip
  port=5432 dbname=postgres" -a readonly_user --schema=public
```

4. Query data through PostgREST:

```
GET http://postgrest_server_ip:3000/table_name
```

Note: table_name must be write in lowercase.