

**Standard** ECMA-348

2<sup>nd</sup> Edition / June 2004

**Web Services  
Description Language  
(WSDL) for CSTA  
Phase III**

**Standard**



Standard  
ECMA-348  
2<sup>nd</sup> Edition / June 2004

---

**Web Services Description  
Language (WSDL) for CSTA  
Phase III**

---



## Brief history

This Standard defines a Web Services Description Language (WSDL) for Phase III of Computer Supported Telecommunications Applications (CSTA).

This Standard is part of a Suite of Standards and Technical Reports for Phase III of CSTA. All of the Standards and Technical Reports in the Suite are based upon the practical experience of Ecma member companies and each one represents a pragmatic and widely based consensus.

This 2nd Edition of ECMA-348 is fully aligned with ECMA-269 6<sup>th</sup> Edition and it imports the CSTA XML schemata from ECMA-323 3<sup>rd</sup> Edition.

This Ecma Standard has been adopted by the General Assembly of June 2004.

---



## **WSDL Tools Acknowledgement**

The WSDL specified in this Standard has been checked using the XML SPY tool.

---





## Table of contents

<b>1</b>	<b>Scope</b>	<b>1</b>
<b>2</b>	<b>Conformance</b>	<b>1</b>
<b>3</b>	<b>References</b>	<b>1</b>
3.1	Ecma References	1
3.2	World Wide Web Consortium References	1
<b>4</b>	<b>Definitions</b>	<b>2</b>
<b>5</b>	<b>Web Services Overview</b>	<b>2</b>
<b>6</b>	<b>WSDL Overview</b>	<b>3</b>
6.1	Namespaces	3
6.2	Data Types	3
6.3	Messages	3
6.3.1	Message Names	4
6.3.2	Message Parts	4
6.3.3	Example	4
6.4	Operations	4
6.4.1	Request-response Operation Type	4
6.4.2	One-way Operation Type	5
6.4.3	Solicit-response Operation Type	5
6.4.4	Notification Operation Type	5
6.4.5	Summary of CSTA Request/Response & Event Models and WSDL Operation Types	6
6.5	Port Types	6
6.5.1	Example	7
6.6	Bindings	8
6.7	Port	8
6.8	Service	8
<b>7</b>	<b>Capability Exchange Services – Messages and Operations</b>	<b>9</b>
7.1	Messages	9
7.2	Operations	12
<b>8</b>	<b>System Services – Messages and Operations</b>	<b>14</b>
8.1	Messages	14

8.2	Operations	18
<b>9</b>	<b>Monitoring Services – Messages and Operations</b>	<b>21</b>
9.1	Messages	21
9.2	Operations	23
<b>10</b>	<b>Snapshot Services – Messages and Operations</b>	<b>24</b>
10.1	Messages	24
10.2	Operations	26
<b>11</b>	<b>Call Control Features – Messages and Operations</b>	<b>27</b>
11.1	Messages	27
11.2	Operations	44
<b>12</b>	<b>Call Associated Features - Messages and Operations</b>	<b>53</b>
12.1	Messages	53
12.2	Operations	57
<b>13</b>	<b>Media Attachment Features - Messages and Operations</b>	<b>59</b>
13.1	Messages	59
13.2	Operations	61
<b>14</b>	<b>Routeing Features - Messages and Operations</b>	<b>62</b>
14.1	Messages	62
14.2	Operations	65
<b>15</b>	<b>Physical Device Features - Messages and Operations</b>	<b>67</b>
15.1	Messages	67
15.2	Operations	80
<b>16</b>	<b>Logical Device Features - Messages and Operations</b>	<b>87</b>
16.1	Messages	87
16.2	Operations	99
<b>17</b>	<b>Device Maintenance Events - Messages and Operations</b>	<b>106</b>
17.1	Messages	106
17.2	Operations	107
<b>18</b>	<b>I/O Services Features - Messages and Operations</b>	<b>108</b>
18.1	Messages	108
18.2	Operations	114

<b>19</b>	<b>Data Collection Features - Messages and Operations</b>	<b>118</b>
19.1	Messages	118
19.2	Operations	122
<b>20</b>	<b>Voice Features - Messages and Operations</b>	<b>124</b>
20.1	Messages	124
20.2	Operations	136
<b>21</b>	<b>Call Data Recording Features - Messages and Operations</b>	<b>143</b>
21.1	Messages	143
21.2	Operations	146
<b>22</b>	<b>Vendor Specific Features - Messages and Operations</b>	<b>148</b>
22.1	Messages	148
22.2	Operations	151
<b>23</b>	<b>Basic Telephony Profile – Port Type</b>	<b>153</b>
<b>24</b>	<b>Routeing Profile - Port Type</b>	<b>156</b>
<b>25</b>	<b>Level 1a Voice Browser Profile - Port Type</b>	<b>157</b>
<b>26</b>	<b>Level 1b Voice Browser Profile - Port Type</b>	<b>159</b>
<b>27</b>	<b>Level 2a Voice Browser Profile - Port Type</b>	<b>161</b>
<b>28</b>	<b>Level 2b Voice Browser Profile - Port Type</b>	<b>164</b>
<b>29</b>	<b>Minimal uaCSTA Call Control Profile – Port Type</b>	<b>167</b>
<b>30</b>	<b>Basic uaCSTA Call Control Profile – Port Type</b>	<b>169</b>
<b>31</b>	<b>Advanced uaCSTA Call Control Profile – Port Type</b>	<b>173</b>
<b>32</b>	<b>Conferencing uaCSTA with Basic Call Control Profile – Port Type</b>	<b>178</b>
<b>33</b>	<b>Conferencing uaCSTA with Advanced Call Control Profile – Port Type</b>	<b>182</b>
<b>34</b>	<b>Basic uaCSTA Device Feature with Basic Call Control Profile – Port Type</b>	<b>188</b>
<b>35</b>	<b>Basic uaCSTA Device Feature with Advanced Call Control Profile – Port Type</b>	<b>193</b>
<b>36</b>	<b>Speaker uaCSTA Device Feature with Basic Call Control Profile – Port Type</b>	<b>199</b>

<b>37</b>	<b>Speaker uaCSTA Device Feature with Advanced Call Control Profile – Port Type</b>	<b>204</b>
	<b>Annex A (informative) Example SOAP 1.1/HTTP Bindings</b>	<b>211</b>
	<b>Annex B (informative) WSDL Considerations</b>	<b>213</b>
	<b>Annex C (informative) Summary of Changes in this Edition</b>	<b>215</b>

## 1 Scope

---

Web Services Description Language (WSDL) is an XML schema for describing Web services and how they can be accessed by Web based applications. WSDL facilitates the creation and deployment of web based applications. For example, by using WSDL with many industry Web services development environments, a web services developer can access features provided by an implementation without knowing details of the network or underlying transport protocols.

This Standard specifies a WSDL for ECMA-269, Services for Computer Supported Telecommunications Applications (CSTA) Phase III. All CSTA features (e.g., services and events) specified in ECMA-269 are specified in this Standard. This Standard specifies port types for all the profiles specified in ECMA-269.

A full WSDL document contains both abstract definitions (WSDL messages and port type elements) and concrete protocol specific definitions (WSDL bindings, ports, and service elements). This CSTA WSDL Standard specifies only the abstract definitions of a WSDL document. The concrete and protocol specific definitions are implementation specific and are outside the scope of this Standard. One example transport illustrated in this Standard is SOAP over HTTP.

WSDL is defined in XML and XML Schemata. This Standard builds upon the XML data types and imports all message formats specified in ECMA-323.

## 2 Conformance

---

In addition to the conformance requirements specified in clause 2 of ECMA-269 and clause 2 of ECMA-323, a conformant implementation supports at least one of the port types defined in this Standard. Each port type in this Standard corresponds to a profile specified in clause 2.3.1 of ECMA-269.

To support one of these port types, an implementation must support all the WSDL operations specified in the port type. A conformant implementation supports an operation by adhering to the corresponding service definition in ECMA-269 and by using the message formats and data types as specified in ECMA-323.

## 3 References

---

### 3.1 Ecma References

- |            |  |
|------------|--|
| ECMA-269   | Services for Computer Supported Telecommunications Applications (CSTA) Phase III, 6 <sup>th</sup> edition (June 2004)                                |
| ECMA-323   | XML Protocol for Computer Supported Telecommunications Applications (CSTA) Phase III, 3 <sup>rd</sup> edition (June 2004)                            |
| ECMA TR/72 | Glossary of Definitions and Terminology for Computer Supported Telecommunications Applications (CSTA) Phase III, 3 <sup>rd</sup> edition (June 2000) |

### 3.2 World Wide Web Consortium References

- |                |  |
|----------------|--|
| SOAP 1.1       | Simple Object Access Protocol (SOAP) 1.1, W3C Note 08 May 2000   |
| XML 1.0        | Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation 06 October 2000  |
| XML Schema 1.0 | XML Schema Language Part 1: Structures. W3C Recommendation 2 May 2001 and XML Schema Language Part 2: Datatypes. W3C Recommendation 2 May 2001 |
| WSDL 1.1       | Web Services Description Language (WSDL) 1.1, W3C Note 15 March 2001   |

## 4 Definitions

---

CSTA-specific terminology is defined in ECMA TR/72.

## 5 Web Services Overview

---

A Web service is the name for a service that is accessed by applications over the Internet. Conceptually this corresponds to a CSTA application (client) invoking services provided by a CSTA implementation (server) in a distributed environment.

Web services are more formally defined as: “encapsulated, loosely coupled contracted functions offered via standard protocols” where:

- Encapsulated means the implementation of the function is never seen from the outside
- Loosely coupled means changing the implementation of one function does not require a change of the invoking function and
- Contracted means there are publicly available descriptions of the function’s behaviour, how to bind to the function as well as its input and output parameters.

The goal of Web services is a seamless integration of services across the different technologies (networks, software, hardware, etc.) available on the Internet. Web services require:

1. A standard way to represent data
2. A common, extensible message format
3. A common, extensible service description language
4. A common way to discover service providers

The first two requirements, a standard way to represent data and a common extensible message format, have already been achieved with ECMA-323. The ECMA-323 Standard includes extensible XML schemas for all of the categories of services defined in ECMA-269.

### NOTE

*This Standard satisfies the third requirement, a common, extensible service description language, by providing a WSDL for CSTA.*

UDDI (Universal Discovery, Description, and Integration Registry) and other procedures to satisfy the fourth requirement are outside the scope of this Standard.

There are three components that are involved with Web services:

- Service Requester – provides the following major functions: 1) optionally queries a Service Broker for where to find a suitable Service Provider and 2) binds and interacts with a Service Provider.
- Service Provider – provides the following major functions: 1) publishes its services (e.g. WSDL document) and 2) interacts with a Service Requester.
- Service Broker – acts as a look up service between a Service Provider and a Service Requester. A Service Broker is an optional component of a Web service.

For the purposes of this Standard:

- A CSTA Service Requester is a Service Requester.
- A CSTA Service Provider is a Service Provider.
- A Service Broker is outside the scope of this Standard.

## 6 WSDL Overview

WSDL enables applications to communicate with each other in an automated way. A WSDL document describes Web services both in an abstract (network independent) and concrete (with network protocol bindings, e.g. SOAP) way.

A WSDL document consists of sets of definitions. There are six major elements in a WSDL document:

- types – provides the data type definitions used to describe the messages exchanged
- message – defines the messages contents
- operation – defines how input and output messages are associated with a logical operation
- portType – groups a set of operations together
- binding – specifies concrete protocol and data format specifications for the operations and messages for a particular portType
- port – specifies an address for a binding
- service – used to group a set of related ports

### 6.1 Namespaces

There are two CSTA specific namespaces used in this Standard:

- <http://www.ecma-international.org/standards/ecma-323/csta/ed3> - this namespace, for which “csta” is a shorthand, contains all of the CSTA XML messages specified in ECMA-323, 3rd edition. See 6.2.
- <http://www.ecma-international.org/standards/ecma-348/csta-wsdl/ed2> - this namespace, for which “tns” is a shorthand, contains the CSTA WSDL schema definitions specified in this Standard.

There are also several other namespaces referenced by the CSTA WSDL schema such as:

- <http://schemas.xmlsoap.org/wsdl/> - this contains the W3C WSDL 1.1 schema
- <http://schemas.xmlsoap.org/wsdl/soap> - this contains the W3C SOAP bindings for WSDL 1.1
- <http://schemas.xmlsoap.org/wsdl/http> - this contains the W3C SOAP bindings for WSDL 1.1
- <http://www.w3.org/2001/XMLSchema> - this contains the W3C XML Schema Definition

### 6.2 Data Types

Data types describe the format of the XML elements in a message. Examples of data types are integers and strings.

This Standard imports the CSTA XML data types standardized in ECMA-323.

### 6.3 Messages

Message definitions specify the contents of an XML message. The message definitions are used in a WSDL operation where they will be typed as an input or output message in the context of the WSDL operation.

This Standard imports the CSTA XML elements specified in ECMA-323.

An example of an XML element specified in ECMA-323 is:

```
<xsd:element name="AnswerCall">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="callToBeAnswered" type="csta:ConnectionID"/>
      <xsd:element ref="csta:correlatorData" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```
<xsd:element ref="csta:userData" minOccurs="0"/>
<xsd:element ref="csta:extensions" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

The above definition, along with all ECMA-323 message definitions, is imported by the CSTA WSDL schema and is available in the <http://www.ecma-international.org/standards/ecma-323/csta/ed3> namespace.

### 6.3.1 Message Names

The WSDL message element specifies the name of a message that is referenced in a WSDL operation element. The naming convention used in this Standard is to derive the WSDL message element name from the corresponding ECMA-323 element name. A lower case letter is used to differentiate the WSDL message name from the ECMA-323 element name.

For example, the ECMA-323 element name AnswerCall is used in the definition of the CSTA WSDL answerCall message name.

### 6.3.2 Message Parts

The WSDL message part element is used to define the actual format of the message. Although there can be multiple parts in a WSDL message, CSTA WSDL uses only one part and therefore can always use the same name (i.e. "parameter") for the message part name. The element type refers to an element defined in ECMA-323.

### 6.3.3 Example

The following example defines a WSDL message element definition for the ECMA-323 AnswerCall request message. Its name "answerCall" will be used to define the AnswerCall Operation.

```
<message name="answerCall">
  <part name="parameter" element="csta:AnswerCall"/>
</message>
```

## 6.4 Operations

A WSDL operation specifies the input message (if any), the output message (if any), and the fault message (if any) that is associated with the WSDL operation.

This formalizes the relationship between the CSTA XML elements standardized in ECMA-323. Although a person can determine the relationship between messages associated with an operation in ECMA-323 by its logical place in the ECMA-323 Standard and in some cases by naming conventions, CSTA WSDL specifies this relationship in a formal notation that can be understood by applications.

The operation name is derived from the feature name as specified in ECMA-323 and the message names are defined by the WSDL message element names.

The following subsections describe the types of WSDL operations used in CSTA WSDL.

### 6.4.1 Request-response Operation Type

A request-response operation type is used when a service provider receives a message and responds with a corresponding message.

It is defined using one input message, one output message, and a fault message.

The CSTA Answer Call service uses this type of operation. The operation for CSTA Answer Call is shown below.

```
<operation name="CSTA-Answer-Call">
  <input message="tns:answerCall"/>
  <output message="tns:answerCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```



The following notation is used to describe a WSDL request-response operation type:

CSTA-Answer-Call	
input	<code>tns:answerCall</code>
output	<code>tns:answerCallResponse</code>
fault	<code>tns:negResponse</code>

#### 6.4.2 One-way Operation Type

A one-way operation type is used when a service provider receives a message and does not respond to the message.

This type of WSDL operation is defined using one input message.

The CSTA-Route-Select service uses this type of operation. The operation for CSTA-Route-Select is shown below.

```
<operation name="CSTA-Route-Select">
  <input message="tns:routeSelect"/>
</operation>
```

The following notation is used to describe a WSDL one-way operation type:

CSTA-Route-Select	
input	<code>tns:routeSelect</code>

WSDL 1.1 does not specify a fault message with a one-way operation type, however ECMA-269 specifies that a negative acknowledgement can be sent for certain services that do not specify a positive acknowledgement. Therefore a separate WSDL operation is specified in this Standard that provides a negative acknowledgement. It is not required to send a negative acknowledgement in these cases.

```
<operation name="CSTA-Fault-One-Way">
  <output message="tns:negResponse"/>
</operation>
```

#### 6.4.3 Solicit-response Operation Type

A solicit-response operation type is used when a service provider sends a message and receives a corresponding response to the message.

It is defined using one output message, one input message, and a fault message.

The CSTA Call Detail Record Report service uses this type of operation. The operation is shown below.

```
<operation name="CSTA-CDR-Report">
  <output message="tns:cDRReport"/>
  <input message="tns:cDRReportResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

The following notation is used to describe a WSDL solicit-response operation type:

CSTA-CDR-Report	
output	<code>tns:cDRReport</code>
input	<code>tns:cDRReportResponse</code>
fault	<code>tns:negResponse</code>

#### 6.4.4 Notification Operation Type

A notification operation type is used when a service provider sends a message and does not receive a response to the message.

This type of WSDL operation is defined using one output message.

The CSTA-Delivered-Event uses this type of operation. The operation for the CSTA-Delivered-Event is shown below.

```
<operation name="CSTA-Delivered-Event">
  <output message="tns:deliveredEvent"/>
</operation>
```

The following notation is used to describe a WSDL notification operation type:

<b>CSTA-Delivered-event</b> output <b>tns:deliveredEvent</b>
---

The CSTA-Route-Request service also uses this type of operation. The operation for CSTA-Route-Request is shown below.

```
<operation name="CSTA-Route-Request">
  <output message="tns:routeRequest"/>
</operation>
```

The following notation is another example of a WSDL notification operation type:

<b>CSTA-Route-Request</b> output <b>tns:routeRequest</b>
---

WSDL 1.1 does not specify a fault message with a notification operation type, however ECMA-269 specifies that a negative acknowledgement can be sent for certain services that do not specify a positive acknowledgement. Therefore a separate WSDL operation is specified in this Standard that provides a negative acknowledgement. It is not required to send a negative acknowledgement in these cases.

```
<operation name="CSTA-Fault-Notification">
  <input message="tns:negResponse"/>
</operation>
```

#### 6.4.5 Summary of CSTA Request/Response & Event Models and WSDL Operation Types

The following list shows how the CSTA request/response models and events are represented with WSDL operation types:

- CSTA Services initiated by a CSTA Service Requester
  - With positive ack. – uses WSDL Request-response operation type.
  - With no positive ack – uses WSDL One-way operation type.
- CSTA Event Reports sent by a CSTA Service Provider
  - Uses WSDL Notification operation type.
- CSTA Services initiated by a CSTA Service Provider
  - With positive ack. – Uses WSDL Solicit-response operation type.
  - With no positive ack – Uses WSDL Notification operation type.

WSDL 1.1 does not allow a fault element on One-way and Notification operation types.

## 6.5 Port Types

A WSDL port type is a collection of WSDL operations. The WSDL port type will be used to bind to one or more specific protocols and transport data formats.

Each port type specified in this Standard (beginning in clause 23) corresponds to a profile specified in clause 2.1.3 of ECMA-269.

A Service Provider may support other operations in addition to the operations specified in the port types.

### 6.5.1 Example

An example port type with three CSTA WSDL operations follows:

```
<portType name="samplePortType">

  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>

  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>

  <operation name="CSTA-Make-Call">
    <input message="tns:makeCall"/>
    <output message="tns:makeCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
</portType>
```

The following notation is used to describe a portType:

porttype **samplePortType**

operations	<b>CSTA-Answer-Call</b> input <b>tns:answerCall</b> output <b>tns:answerCallResponse</b> fault <b>tns:negResponse</b>
	<b>CSTA-Clear-Connection</b> input <b>tns:clearConnection</b> output <b>tns:clearConnectionResponse</b> fault <b>tns:negResponse</b>
	<b>CSTA-Make-Call</b> input <b>tns:makeCall</b> output <b>tns:makeCallResponse</b> fault <b>tns:negResponse</b>
source	<pre>&lt;portType name="samplePortType"&gt;   &lt;operation name="CSTA-Answer-Call"&gt;     &lt;input message="tns:answerCall"/&gt;     &lt;output message="tns:answerCallResponse"/&gt;     &lt;fault name="FaultName" message="tns:negResponse"/&gt;   &lt;/operation&gt;</pre>

```
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 6.6 Bindings

A binding associates one concrete protocol and transport data format specification to a port type.

This Standard allows any type of binding and binding extensions for the CSTA WSDL port types specified in this Standard.

### NOTE

*Although the base WSDL 1.1 specification supports all of the WSDL operation types defined in 6.3, WSDL 1.1 only defines bindings for Request-response and One-way operation types. Binding extensions are necessary to support the CSTA WSDL Solicit-response and Notification operation types defined in this Standard.*

An example of a SOAP 1.1/HTTP binding is provided in Annex A.

## 6.7 Port

A port is a single endpoint defined as a combination of a binding and a network address.

An example of how a port is used in a WSDL document is provided in Annex A.

## 6.8 Service

A service is a collection of related ports.

An example of how a service is used in a WSDL document is provided in Annex A.

## 7 Capability Exchange Services – Messages and Operations

This clause specifies the WSDL messages and operations for the CSTA capability exchange services and messages.

### 7.1 Messages

#### message **getCSTAFeatures**

```
parts      parameter
           element  csta:GetCSTAFeatures
source    <message name="getCSTAFeatures">
           <part name="parameter" element="csta:GetCSTAFeatures"/>
           </message>
```

#### message **getCSTAFeaturesResponse**

```
parts      parameter
           element  csta:GetCSTAFeaturesResponse
source    <message name="getCSTAFeaturesResponse">
           <part name="parameter" element="csta:GetCSTAFeaturesResponse"/>
           </message>
```

#### message **getLogicalDeviceInformation**

```
parts      parameter
           element  csta:GetLogicalDeviceInformation
source    <message name="getLogicalDeviceInformation">
           <part name="parameter" element="csta:GetLogicalDeviceInformation"/>
           </message>
```

#### message **getLogicalDeviceInformationResponse**

```
parts      parameter
           element  csta:GetLogicalDeviceInformationResponse
source    <message name="getLogicalDeviceInformationResponse">
           <part name="parameter" element="csta:GetLogicalDeviceInformationResponse"/>
           </message>
```

### message **getPhysicalDeviceInformation**

```
parts    parameter
         element    csta:GetPhysicalDeviceInformation
source   <message name="getPhysicalDeviceInformation">
         <part name="parameter" element="csta:GetPhysicalDeviceInformation"/>
         </message>
```

### message **getPhysicalDeviceInformationResponse**

```
parts    parameter
         element    csta:GetPhysicalDeviceInformationResponse
source   <message name="getPhysicalDeviceInformationResponse">
         <part name="parameter" element="csta:GetPhysicalDeviceInformationResponse"/>
         </message>
```

### message **getSwitchingFunctionCaps**

```
parts    parameter
         element    csta:GetSwitchingFunctionCaps
source   <message name="getSwitchingFunctionCaps">
         <part name="parameter" element="csta:GetSwitchingFunctionCaps"/>
         </message>
```

### message **getSwitchingFunctionCapsResponse**

```
parts    parameter
         element    csta:GetSwitchingFunctionCapsResponse
source   <message name="getSwitchingFunctionCapsResponse">
         <part name="parameter" element="csta:GetSwitchingFunctionCapsResponse"/>
         </message>
```

### message **getSwitchingFunctionDevices**

```
parts    parameter
        element csta:GetSwitchingFunctionDevices
source   <message name="getSwitchingFunctionDevices">
        <part name="parameter" element="csta:GetSwitchingFunctionDevices"/>
        </message>
```

### message **getSwitchingFunctionDevicesResponse**

```
parts    parameter
        element csta:GetSwitchingFunctionDevicesResponse
source   <message name="getSwitchingFunctionDevicesResponse">
        <part name="parameter" element="csta:GetSwitchingFunctionDevicesResponse"/>
        </message>
```

### message **switchingFunctionDevices**

```
parts    parameter
        element csta:SwitchingFunctionDevices
source   <message name="switchingFunctionDevices">
        <part name="parameter" element="csta:SwitchingFunctionDevices"/>
        </message>
```

## 7.2 Operations

operations

### CSTA-Get-CSTA-Features

input [tns:getCSTAFeatures](#)  
 output [tns:getCSTAFeaturesResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Get-Logical-Device-Information

input [tns:getLogicalDeviceInformation](#)  
 output [tns:getLogicalDeviceInformationResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Get-Physical-Device-Information

input [tns:getPhysicalDeviceInformation](#)  
 output [tns:getPhysicalDeviceInformationResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Get-Switching-Function-Capabilities

input [tns:getSwitchingFunctionCaps](#)  
 output [tns:getSwitchingFunctionCapsResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Get-Switching-Function-Devices

input [tns:getSwitchingFunctionDevices](#)  
 output [tns:getSwitchingFunctionDevicesResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Switching-Function-Devices

output [tns:switchingFunctionDevices](#)  
 source 

```
<operation name="CSTA-Get-CSTA-Features">
  <input message="tns:getCSTAFeatures"/>
  <output message="tns:getCSTAFeaturesResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Logical-Device-Information">
  <input message="tns:getLogicalDeviceInformation"/>
  <output message="tns:getLogicalDeviceInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Physical-Device-Information">
  <input message="tns:getPhysicalDeviceInformation"/>
  <output message="tns:getPhysicalDeviceInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```



```
<operation name="CSTA-Get-Switching-Function-Capabilities">  
  <input message="tns:getSwitchingFunctionCaps"/>  
  <output message="tns:getSwitchingFunctionCapsResponse"/>  
  <fault name="FaultName" message="tns:negResponse"/>  
</operation>  
<operation name="CSTA-Get-Switching-Function-Devices">  
  <input message="tns:getSwitchingFunctionDevices"/>  
  <output message="tns:getSwitchingFunctionDevicesResponse"/>  
  <fault name="FaultName" message="tns:negResponse"/>  
</operation>  
<operation name="CSTA-Switching-Function-Devices">  
  <output message="tns:switchingFunctionDevices"/>  
</operation>
```

## 8 System Services – Messages and Operations

This clause specifies the WSDL messages and operations for the CSTA system services.

### 8.1 Messages

#### message **changeSystemStatFilter**

```
parts      parameter
           element   csta:ChangeSysStatFilter
source     <message name="changeSystemStatFilter">
           <part name="parameter" element="csta:ChangeSysStatFilter"/>
           </message>
```

#### message **changeSysStatFilterResponse**

```
parts      parameter
           element   csta:ChangeSysStatFilterResponse
source     <message name="changeSysStatFilterResponse">
           <part name="parameter" element="csta:ChangeSysStatFilterResponse"/>
           </message>
```

#### message **systemRegister**

```
parts      parameter
           element   csta:SystemRegister
source     <message name="systemRegister">
           <part name="parameter" element="csta:SystemRegister"/>
           </message>
```

#### message **systemRegisterResponse**

```
parts      parameter
           element   csta:SystemRegisterResponse
source     <message name="systemRegisterResponse">
           <part name="parameter" element="csta:SystemRegisterResponse"/>
           </message>
```

### message **systemRegisterAbort**

```
parts    parameter
         element  csta:SystemRegisterAbort
source   <message name="systemRegisterAbort">
         <part name="parameter" element="csta:SystemRegisterAbort"/>
         </message>
```

### message **systemRegisterCancel**

```
parts    parameter
         element  csta:SystemRegisterCancel
source   <message name="systemRegisterCancel">
         <part name="parameter" element="csta:SystemRegisterCancel"/>
         </message>
```

### message **systemRegisterCancelResponse**

```
parts    parameter
         element  csta:SystemRegisterCancelResponse
source   <message name="systemRegisterCancelResponse">
         <part name="parameter" element="csta:SystemRegisterCancelResponse"/>
         </message>
```

### message **requestSystemStatus**

```
parts    parameter
         element  csta:RequestSystemStatus
source   <message name="requestSystemStatus">
         <part name="parameter" element="csta:RequestSystemStatus"/>
         </message>
```

### message **requestSystemStatusResponse**

```
parts    parameter
         element  csta:RequestSystemStatusResponse
source   <message name="requestSystemStatusResponse">
         <part name="parameter" element="csta:RequestSystemStatusResponse"/>
         </message>
```

### message **systemStatus**

```
parts    parameter
         element    csta:SystemStatus
source  <message name="systemStatus">
         <part name="parameter" element="csta:SystemStatus"/>
         </message>
```

### message **systemStatusResponse**

```
parts    parameter
         element    csta:SystemStatusResponse
source  <message name="systemStatusResponse">
         <part name="parameter" element="csta:SystemStatusResponse"/>
         </message>
```

### message **swFunctionCapsChanged**

```
parts    parameter
         element    csta:SwFunctionCapsChanged
source  <message name="swFunctionCapsChanged">
         <part name="parameter" element="csta:SwFunctionCapsChanged"/>
         </message>
```

### message **swFunctionCapsChangedResponse**

```
parts    parameter
         element    csta:SwFunctionCapsChangedResponse
source  <message name="swFunctionCapsChangedResponse">
         <part name="parameter" element="csta:SwFunctionCapsChangedResponse"/>
         </message>
```

### message **swFunctionDevicesChanged**

```
parts    parameter
         element csta:SwFunctionDevicesChanged
source  <message name="swFunctionDevicesChanged">
        <part name="parameter" element="csta:SwFunctionDevicesChanged"/>
        </message>
```

### message **swFunctionDevicesChangedResponse**

```
parts    parameter
         element csta:SwFunctionDevicesChangedResponse
source  <message name="swFunctionDevicesChangedResponse">
        <part name="parameter" element="csta:SwFunctionDevicesChangedResponse"/>
        </message>
```

## 8.2 Operations

operations

### CSTA-Change-System-Status-Filter

input [tns:changeSystemStatFilter](#)  
output [tns:changeSysStatFilterResponse](#)  
fault [tns:negResponse](#)

### CSTA-System-Register

input [tns:systemRegister](#)  
output [tns:systemRegisterResponse](#)  
fault [tns:negResponse](#)

### CSTA-System-Register-Abort

output [tns:systemRegisterAbort](#)

### CSTA-System-Register-Cancel

input [tns:systemRegisterCancel](#)  
output [tns:systemRegisterCancelResponse](#)  
fault [tns:negResponse](#)

### CSTA-Request-System-Status-as-input

input [tns:requestSystemStatus](#)  
output [tns:requestSystemStatusResponse](#)  
fault [tns:negResponse](#)

### CSTA-Request-System-Status-as-output

output [tns:requestSystemStatus](#)  
input [tns:requestSystemStatusResponse](#)  
fault [tns:negResponse](#)

### CSTA-System-Status-as-input

input [tns:systemStatus](#)  
output [tns:systemStatusResponse](#)  
fault [tns:negResponse](#)

### CSTA-System-Status-as-output

output [tns:systemStatus](#)  
input [tns:systemStatusResponse](#)  
fault [tns:negResponse](#)

### CSTA-SF-Capabilities-Changed

output [tns:swFunctionCapsChanged](#)  
input [tns:swFunctionCapsChangedResponse](#)  
fault [tns:negResponse](#)

### CSTA-SF-Devices-Changed

```
    output tns:swFunctionDevicesChanged
    input  tns:swFunctionDevicesChangedResponse
    fault  tns:negResponse
source <operation name="CSTA-Change-System-Status-Filter">
  <input message="tns:changeSystemStatFilter"/>
  <output message="tns:changeSysStatFilterResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-System-Register">
  <input message="tns:systemRegister"/>
  <output message="tns:systemRegisterResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-System-Register-Abort">
  <output message="tns:systemRegisterAbort"/>
</operation>
<operation name="CSTA-System-Register-Cancel">
  <input message="tns:systemRegisterCancel"/>
  <output message="tns:systemRegisterCancelResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Request-System-Status-as-input">
  <input message="tns:requestSystemStatus"/>
  <output message="tns:requestSystemStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Request-System-Status-as-output">
  <output message="tns:requestSystemStatus"/>
  <input message="tns:requestSystemStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-System-Status-as-input">
  <input message="tns:systemStatus"/>
  <output message="tns:systemStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-System-Status-as-output">
  <output message="tns:systemStatus"/>
  <input message="tns:systemStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-SF-Capabilities-Changed">
```

```
<output message="tns:swFunctionCapsChanged"/>
<input message="tns:swFunctionCapsChangedResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-SF-Devices-Changed">
  <output message="tns:swFunctionDevicesChanged"/>
  <input message="tns:swFunctionDevicesChangedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```



## 9 Monitoring Services – Messages and Operations

This clause specifies the WSDL messages and operations for the CSTA monitoring services.

### 9.1 Messages

#### message **changeMonitorFilter**

```
parts    parameter
         element    csta:ChangeMonitorFilter
source  <message name="changeMonitorFilter">
         <part name="parameter" element="csta:ChangeMonitorFilter"/>
         </message>
```

#### message **changeMonitorFilterResponse**

```
parts    parameter
         element    csta:ChangeMonitorFilterResponse
source  <message name="changeMonitorFilterResponse">
         <part name="parameter" element="csta:ChangeMonitorFilterResponse"/>
         </message>
```

#### message **monitorStart**

```
parts    parameter
         element    csta:MonitorStart
source  <message name="monitorStart">
         <part name="parameter" element="csta:MonitorStart"/>
         </message>
```

#### message **monitorStartResponse**

```
parts    parameter
         element    csta:MonitorStartResponse
source  <message name="monitorStartResponse">
         <part name="parameter" element="csta:MonitorStartResponse"/>
         </message>
```

### message **monitorStop**

parts	<b>parameter</b>
	element <b>csta:MonitorStop</b>
source	<code>&lt;message name="monitorStop"&gt;</code> <code>&lt;part name="parameter" element="csta:MonitorStop"/&gt;</code> <code>&lt;/message&gt;</code>

### message **monitorStopResponse**

parts	<b>parameter</b>
	element <b>csta:MonitorStopResponse</b>
source	<code>&lt;message name="monitorStopResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:MonitorStopResponse"/&gt;</code> <code>&lt;/message&gt;</code>

## 9.2 Operations

operations	<p><b>CSTA-Change-Monitor-Filter</b></p> <p>input    <a href="#"><u>tns:changeMonitorFilter</u></a></p> <p>output   <a href="#"><u>tns:changeMonitorFilterResponse</u></a></p> <p>fault    <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Start</b></p> <p>input    <a href="#"><u>tns:monitorStart</u></a></p> <p>output   <a href="#"><u>tns:monitorStartResponse</u></a></p> <p>fault    <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Stop-as-input</b></p> <p>input    <a href="#"><u>tns:monitorStop</u></a></p> <p>output   <a href="#"><u>tns:monitorStopResponse</u></a></p> <p>fault    <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Stop-as-output</b></p> <p>input    <a href="#"><u>tns:monitorStop</u></a></p> <p>output   <a href="#"><u>tns:monitorStopResponse</u></a></p> <p>fault    <a href="#"><u>tns:negResponse</u></a></p>
source	<pre> &lt;operation name="CSTA-Change-Monitor-Filter"&gt;   &lt;input message="tns:changeMonitorFilter"/&gt;   &lt;output message="tns:changeMonitorFilterResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt;  &lt;operation name="CSTA-Monitor-Start"&gt;   &lt;input message="tns:monitorStart"/&gt;   &lt;output message="tns:monitorStartResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt;  &lt;operation name="CSTA-Monitor-Stop-as-input"&gt;   &lt;input message="tns:monitorStop"/&gt;   &lt;output message="tns:monitorStopResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt;  &lt;operation name="CSTA-Monitor-Stop-as-output"&gt;   &lt;output message="tns:monitorStop"/&gt;   &lt;input message="tns:monitorStopResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; </pre>

## 10 Snapshot Services – Messages and Operations

This clause specifies the WSDL messages and operations for the CSTA snapshot services.

### 10.1 Messages

#### message **snapshotCall**

```
parts    parameter
         element    csta:SnapshotCall
source  <message name="snapshotCall">
         <part name="parameter" element="csta:SnapshotCall"/>
       </message>
```

#### message **snapshotCallResponse**

```
parts    parameter
         element    csta:SnapshotCallResponse
source  <message name="snapshotCallResponse">
         <part name="parameter" element="csta:SnapshotCallResponse"/>
       </message>
```

#### message **snapshotDevice**

```
parts    parameter
         element    csta:SnapshotDevice
source  <message name="snapshotDevice">
         <part name="parameter" element="csta:SnapshotDevice"/>
       </message>
```

#### message **snapshotDeviceResponse**

```
parts    parameter
         element    csta:SnapshotDeviceResponse
source  <message name="snapshotDeviceResponse">
         <part name="parameter" element="csta:SnapshotDeviceResponse"/>
       </message>
```

### message **snapshotCallData**

```
parts    parameter
         element  csta:SnapshotCallData
source  <message name="snapshotCallData">
        <part name="parameter" element="csta:SnapshotCallData"/>
        </message>
```

### message **snapshotDeviceData**

```
parts    parameter
         element  csta:SnapshotDeviceData
source  <message name="snapshotDeviceData">
        <part name="parameter" element="csta:SnapshotDeviceData"/>
        </message>
```

## 10.2 Operations

operations	<p><b>CSTA-Snapshot-Call</b></p> <p>input    <a href="#">tns:snapshotCall</a></p> <p>output   <a href="#">tns:snapshotCallResponse</a></p> <p>fault    <a href="#">tns:negResponse</a></p> <p><b>CSTA-Snapshot-Device</b></p> <p>input    <a href="#">tns:snapshotDevice</a></p> <p>output   <a href="#">tns:snapshotDeviceResponse</a></p> <p>fault    <a href="#">tns:negResponse</a></p> <p><b>CSTA-Snapshot-Call-Data</b></p> <p>output   <a href="#">tns:snapshotCallData</a></p> <p><b>CSTA-Snapshot-Device-Data</b></p> <p>output   <a href="#">tns:snapshotDeviceData</a></p>
source	<pre> &lt;operation name="CSTA-Snapshot-Call"&gt;   &lt;input message="tns:snapshotCall"/&gt;   &lt;output message="tns:snapshotCallResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Snapshot-Device"&gt;   &lt;input message="tns:snapshotDevice"/&gt;   &lt;output message="tns:snapshotDeviceResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Snapshot-Call-Data"&gt;   &lt;output message="tns:snapshotCallData"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Snapshot-Device-Data"&gt;   &lt;output message="tns:snapshotDeviceData"/&gt; &lt;/operation&gt; </pre>

## 11 Call Control Features – Messages and Operations

This clause specifies the WSDL messages and operations for the CSTA call control features.

### 11.1 Messages

#### message **acceptCall**

```
parts    parameter
         element  csta:AcceptCall
source   <message name="acceptCall">
         <part name="parameter" element="csta:AcceptCall"/>
         </message>
```

#### message **acceptCallResponse**

```
parts    parameter
         element  csta:AcceptCallResponse
source   <message name="acceptCallResponse">
         <part name="parameter" element="csta:AcceptCallResponse"/>
         </message>
```

#### message **alternateCall**

```
parts    parameter
         element  csta:AlternateCall
source   <message name="alternateCall">
         <part name="parameter" element="csta:AlternateCall"/>
         </message>
```

#### message **alternateCallResponse**

```
parts    parameter
         element  csta:AlternateCallResponse
source   <message name="alternateCallResponse">
         <part name="parameter" element="csta:AlternateCallResponse"/>
         </message>
```

### message **answerCall**

parts	<b>parameter</b>
	element <b>csta:AnswerCall</b>
source	<code>&lt;message name="answerCall"&gt;</code> <code>&lt;part name="parameter" element="csta:AnswerCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **answerCallResponse**

parts	<b>parameter</b>
	element <b>csta:AnswerCallResponse</b>
source	<code>&lt;message name="answerCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:AnswerCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callBack**

parts	<b>parameter</b>
	element <b>csta:CallBack</b>
source	<code>&lt;message name="callBack"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBack"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callBackResponse**

parts	<b>parameter</b>
	element <b>csta:CallBackResponse</b>
source	<code>&lt;message name="callBackResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBackResponse"/&gt;</code> <code>&lt;/message&gt;</code>



### message **callBackMessage**

parts	<b>parameter</b>
	element <b>csta:CallBackMessage</b>
source	<code>&lt;message name="callBackMessage"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBackMessage"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callBackMessageResponse**

parts	<b>parameter</b>
	element <b>csta:CallBackMessageResponse</b>
source	<code>&lt;message name="callBackMessageResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBackMessageResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **camponCall**

parts	<b>parameter</b>
	element <b>csta:CamponCall</b>
source	<code>&lt;message name="camponCall"&gt;</code> <code>&lt;part name="parameter" element="csta:CamponCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **camponCallResponse**

parts	<b>parameter</b>
	element <b>csta:CamponCallResponse</b>
source	<code>&lt;message name="camponCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:CamponCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **clearCall**

parts	<b>parameter</b>
	element <b>csta:ClearCall</b>
source	<code>&lt;message name="clearCall"&gt;</code> <code>&lt;part name="parameter" element="csta:ClearCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **clearCallResponse**

parts	<b>parameter</b>
	element <b>csta:ClearCallResponse</b>
source	<code>&lt;message name="clearCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ClearCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **clearConnection**

parts	<b>parameter</b>
	element <b>csta:ClearConnection</b>
source	<code>&lt;message name="clearConnection"&gt;</code> <code>&lt;part name="parameter" element="csta:ClearConnection"/&gt;</code> <code>&lt;/message&gt;</code>

### message **clearConnectionResponse**

parts	<b>parameter</b>
	element <b>csta:ClearConnectionResponse</b>
source	<code>&lt;message name="clearConnectionResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ClearConnectionResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **conferenceCall**

parts	<b>parameter</b>
	element <b>csta:ConferenceCall</b>
source	<code>&lt;message name="conferenceCall"&gt;</code> <code>&lt;part name="parameter" element="csta:ConferenceCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **conferenceCallResponse**

parts	<b>parameter</b>
	element <b>csta:ConferenceCallResponse</b>
source	<code>&lt;message name="conferenceCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ConferenceCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **consultationCall**

parts	<b>parameter</b>
	element <b>csta:ConsultationCall</b>
source	<code>&lt;message name="consultationCall"&gt;</code> <code>&lt;part name="parameter" element="csta:ConsultationCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **consultationCallResponse**

parts	<b>parameter</b>
	element <b>csta:ConsultationCallResponse</b>
source	<code>&lt;message name="consultationCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ConsultationCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **deflectCall**

parts	<b>parameter</b>
	element <b>csta:DeflectCall</b>
source	<code>&lt;message name="deflectCall"&gt;</code> <code>&lt;part name="parameter" element="csta:DeflectCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **deflectCallResponse**

parts	<b>parameter</b>
	element <b>csta:DeflectCallResponse</b>
source	<code>&lt;message name="deflectCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:DeflectCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **dialDigits**

parts	<b>parameter</b>
	element <b>csta:DialDigits</b>
source	<code>&lt;message name="dialDigits"&gt;</code> <code>&lt;part name="parameter" element="csta:DialDigits"/&gt;</code> <code>&lt;/message&gt;</code>

### message **dialDigitsResponse**

parts	<b>parameter</b>
	element <b>csta:DialDigitsResponse</b>
source	<code>&lt;message name="dialDigitsResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:DialDigitsResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **directedPickupCall**

```
parts    parameter
        element csta:DirectedPickupCall
source  <message name="directedPickupCall">
        <part name="parameter" element="csta:DirectedPickupCall"/>
        </message>
```

### message **directedPickupCallResponse**

```
parts    parameter
        element csta:DirectedPickupCallResponse
source  <message name="directedPickupCallResponse">
        <part name="parameter" element="csta:DirectedPickupCallResponse"/>
        </message>
```

### message **groupPickupCall**

```
parts    parameter
        element csta:GroupPickupCall
source  <message name="groupPickupCall">
        <part name="parameter" element="csta:GroupPickupCall"/>
        </message>
```

### message **groupPickupCallResponse**

```
parts    parameter
        element csta:GroupPickupCallResponse
source  <message name="groupPickupCallResponse">
        <part name="parameter" element="csta:GroupPickupCallResponse"/>
        </message>
```

### message **holdCall**

```
parts    parameter
        element csta:HoldCall
source  <message name="holdCall">
        <part name="parameter" element="csta:HoldCall"/>
        </message>
```

### message holdCallResponse

```
parts    parameter
         element    csta:HoldCallResponse
source  <message name="holdCallResponse">
        <part name="parameter" element="csta:HoldCallResponse"/>
        </message>
```

### message intrudeCall

```
parts    parameter
         element    csta:IntrudeCall
source  <message name="intrudeCall">
        <part name="parameter" element="csta:IntrudeCall"/>
        </message>
```

### message intrudeCallResponse

```
parts    parameter
         element    csta:IntrudeCallResponse
source  <message name="intrudeCallResponse">
        <part name="parameter" element="csta:IntrudeCallResponse"/>
        </message>
```

### message joinCall

```
parts    parameter
         element    csta:JoinCall
source  <message name="joinCall">
        <part name="parameter" element="csta:JoinCall"/>
        </message>
```

### message joinCallResponse

```
parts    parameter
         element  csta:JoinCallResponse
source   <message name="joinCallResponse">
         <part name="parameter" element="csta:JoinCallResponse"/>
         </message>
```

### message makeCall

```
parts    parameter
         element  csta:MakeCall
source   <message name="makeCall">
         <part name="parameter" element="csta:MakeCall"/>
         </message>
```

### message makeCallResponse

```
parts    parameter
         element  csta:MakeCallResponse
source   <message name="makeCallResponse">
         <part name="parameter" element="csta:MakeCallResponse"/>
         </message>
```

### message makePredictiveCall

```
parts    parameter
         element  csta:MakePredictiveCall
source   <message name="makePredictiveCall">
         <part name="parameter" element="csta:MakePredictiveCall"/>
         </message>
```

### message makePredictiveCallResponse

```
parts    parameter
         element  csta:MakePredictiveCallResponse
source   <message name="makePredictiveCallResponse">
         <part name="parameter" element="csta:MakePredictiveCallResponse"/>
         </message>
```

### message **parkCall**

```
parts    parameter
        element csta:ParkCall
source  <message name="parkCall">
        <part name="parameter" element="csta:ParkCall"/>
</message>
```

### message **parkCallResponse**

```
parts    parameter
        element csta:ParkCallResponse
source  <message name="parkCallResponse">
        <part name="parameter" element="csta:ParkCallResponse"/>
</message>
```

### message **reconnectCall**

```
parts    parameter
        element csta:ReconnectCall
source  <message name="reconnectCall">
        <part name="parameter" element="csta:ReconnectCall"/>
</message>
```

### message **reconnectCallResponse**

```
parts    parameter
        element csta:ReconnectCallResponse
source  <message name="reconnectCallResponse">
        <part name="parameter" element="csta:ReconnectCallResponse"/>
</message>
```



### message **retrieveCall**

```
parts    parameter
         element  csta:RetrieveCall
source  <message name="retrieveCall">
         <part name="parameter" element="csta:RetrieveCall"/>
         </message>
```

### message **retrieveCallResponse**

```
parts    parameter
         element  csta:RetrieveCallResponse
source  <message name="retrieveCallResponse">
         <part name="parameter" element="csta:RetrieveCallResponse"/>
         </message>
```

### message **sendMessage**

```
parts    parameter
         element  csta:SendMessage
source  <message name="sendMessage">
         <part name="parameter" element="csta:SendMessage"/>
         </message>
```

### message **sendMessageResponse**

```
parts    parameter
         element  csta:SendMessageResponse
source  <message name="sendMessageResponse">
         <part name="parameter" element="csta:SendMessageResponse"/>
         </message>
```

### message **singleStepConferenceCall**

```
parts    parameter
         element  csta:SingleStepConferenceCall
source  <message name="singleStepConferenceCall">
         <part name="parameter" element="csta:SingleStepConferenceCall"/>
         </message>
```

### message **singleStepConferenceCallResponse**

parts	<b>parameter</b>
	element <b>csta:SingleStepConferenceCallResponse</b>
source	<code>&lt;message name="singleStepConferenceCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SingleStepConferenceCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **singleStepTransferCall**

parts	<b>parameter</b>
	element <b>csta:SingleStepTransferCall</b>
source	<code>&lt;message name="singleStepTransferCall"&gt;</code> <code>&lt;part name="parameter" element="csta:SingleStepTransferCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message **singleStepTransferCallResponse**

parts	<b>parameter</b>
	element <b>csta:SingleStepTransferCallResponse</b>
source	<code>&lt;message name="singleStepTransferCallResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SingleStepTransferCallResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **transferCall**

parts	<b>parameter</b>
	element <b>csta:TransferCall</b>
source	<code>&lt;message name="transferCall"&gt;</code> <code>&lt;part name="parameter" element="csta:TransferCall"/&gt;</code> <code>&lt;/message&gt;</code>

### message transferCallResponse

```
parts    parameter
         element  csta:TransferCallResponse
source  <message name="transferCallResponse">
         <part name="parameter" element="csta:TransferCallResponse"/>
         </message>
```

### message bridgedEvent

```
parts    parameter
         element  csta:BridgedEvent
source  <message name="bridgedEvent">
         <part name="parameter" element="csta:BridgedEvent"/>
         </message>
```

### message callClearedEvent

```
parts    parameter
         element  csta:CallClearedEvent
source  <message name="callClearedEvent">
         <part name="parameter" element="csta:CallClearedEvent"/>
         </message>
```

### message conferencedEvent

```
parts    parameter
         element  csta:ConferencedEvent
source  <message name="conferencedEvent">
         <part name="parameter" element="csta:ConferencedEvent"/>
         </message>
```

### message connectionClearedEvent

```
parts    parameter
         element  csta:ConnectionClearedEvent
source  <message name="connectionClearedEvent">
         <part name="parameter" element="csta:ConnectionClearedEvent"/>
         </message>
```

### message **deliveredEvent**

```
parts    parameter
         element    csta:DeliveredEvent
source  <message name="deliveredEvent">
         <part name="parameter" element="csta:DeliveredEvent"/>
         </message>
```

### message **digitsDialedEvent**

```
parts    parameter
         element    csta:DigitsDialedEvent
source  <message name="digitsDialedEvent">
         <part name="parameter" element="csta:DigitsDialedEvent"/>
         </message>
```

### message **divertedEvent**

```
parts    parameter
         element    csta:DivertedEvent
source  <message name="divertedEvent">
         <part name="parameter" element="csta:DivertedEvent"/>
         </message>
```

### message **establishedEvent**

```
parts    parameter
         element    csta:EstablishedEvent
source  <message name="establishedEvent">
         <part name="parameter" element="csta:EstablishedEvent"/>
         </message>
```

### message failedEvent

```
parts    parameter
         element  csta:FailedEvent
source   <message name="failedEvent">
         <part name="parameter" element="csta:FailedEvent"/>
         </message>
```

### message heldEvent

```
parts    parameter
         element  csta:HeldEvent
source   <message name="heldEvent">
         <part name="parameter" element="csta:HeldEvent"/>
         </message>
```

### message networkCapabilitiesChangedEvent

```
parts    parameter
         element  csta:NetworkCapabilitiesChangedEvent
source   <message name="networkCapabilitiesChangedEvent">
         <part name="parameter" element="csta:NetworkCapabilitiesChangedEvent"/>
         </message>
```

### message networkReachedEvent

```
parts    parameter
         element  csta:NetworkReachedEvent
source   <message name="networkReachedEvent">
         <part name="parameter" element="csta:NetworkReachedEvent"/>
         </message>
```

### message offeredEvent

```
parts    parameter
         element  csta:OfferedEvent
source   <message name="offeredEvent">
         <part name="parameter" element="csta:OfferedEvent"/>
         </message>
```

### message **originatedEvent**

```
parts    parameter
         element csta:OriginatedEvent
source  <message name="originatedEvent">
        <part name="parameter" element="csta:OriginatedEvent"/>
        </message>
```

### message **queuedEvent**

```
parts    parameter
         element csta:QueuedEvent
source  <message name="queuedEvent">
        <part name="parameter" element="csta:QueuedEvent"/>
        </message>
```

### message **retrievedEvent**

```
parts    parameter
         element csta:RetrievedEvent
source  <message name="retrievedEvent">
        <part name="parameter" element="csta:RetrievedEvent"/>
        </message>
```

### message **serviceInitiatedEvent**

```
parts    parameter
         element csta:ServiceInitiatedEvent
source  <message name="serviceInitiatedEvent">
        <part name="parameter" element="csta:ServiceInitiatedEvent"/>
        </message>
```



## message **transferredEvent**

parts	<b>parameter</b>
	element <b>csta:TransferredEvent</b>
source	<code>&lt;message name="transferredEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:TransferredEvent"/&gt;</code> <code>&lt;/message&gt;</code>

## 11.2 Operations

operations

### **CSTA-Accept-Call**

input [tns:acceptCall](#)  
output [tns:acceptCallResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Alternate-Call**

input [tns:alternateCall](#)  
output [tns:alternateCallResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Answer-Call**

input [tns:answerCall](#)  
output [tns:answerCallResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Call-Back-Call-Related**

input [tns:callBack](#)  
output [tns:callBackResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Call-Back-Message-Call-Related**

input [tns:callBackMessage](#)  
output [tns:callBackMessageResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Camp-On-Call**

input [tns:camponCall](#)  
output [tns:camponCallResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Clear-Call**

input [tns:clearCall](#)  
output [tns:clearCallResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Clear-Connection**

input [tns:clearConnection](#)  
output [tns:clearConnectionResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Conference-Call**

input [tns:conferenceCall](#)  
output [tns:conferenceCallResponse](#)  
fault [tns:negResponse](#)



#### CSTA-Consultation-Call

input [tns:consultationCall](#)  
output [tns:consultationCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Deflect-Call

input [tns:deflectCall](#)  
output [tns:deflectCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Dial-Digits-Call

input [tns:dialDigits](#)  
output [tns:dialDigitsResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Directed-Pickup-Call

input [tns:directedPickupCall](#)  
output [tns:directedPickupCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Group-Pickup-Call

input [tns:groupPickupCall](#)  
output [tns:groupPickupCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Hold-Call

input [tns:holdCall](#)  
output [tns:holdCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Intrude-Call

input [tns:intrudeCall](#)  
output [tns:intrudeCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Join-Call

input [tns:joinCall](#)  
output [tns:joinCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Make-Call

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Make-Predictive-Call**

input [tns:makePredictiveCall](#)  
output [tns:makePredictiveCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Park-Call**

input [tns:parkCall](#)  
output [tns:parkCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Reconnect-Call**

input [tns:reconnectCall](#)  
output [tns:reconnectCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Send-Message**

input [tns:sendMessage](#)  
output [tns:sendMessageResponse](#)  
fault [tns:negResponse](#)

**CSTA-Single-Step-Conference-Call**

input [tns:singleStepConferenceCall](#)  
output [tns:singleStepConferenceCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Transfer-Call**

input [tns:transferCall](#)  
output [tns:transferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Bridged-event**

output [tns:bridgedEvent](#)

**CSTA-Call-Cleared-event**

output [tns:callClearedEvent](#)

**CSTA-Conferenced-event**

output [tns:conferencedEvent](#)

**CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Digits-Dialed-event**

output [tns:digitsDialedEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Capabilities-Changed-event**

output [tns:networkCapabilitiesChangedEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Offered-event**

output [tns:offeredEvent](#)

**CSTA-Originated-event**

output [tns:originatedEvent](#)

**CSTA-Queued-event**

output [tns:queuedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

### CSTA-Transferred-event

```
source      output  tns:transferredEvent
<operation name="CSTA-Accept-Call">
  <input message="tns:acceptCall"/>
  <output message="tns:acceptCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Alternate-Call">
  <input message="tns:alternateCall"/>
  <output message="tns:alternateCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Answer-Call">
  <input message="tns:answerCall"/>
  <output message="tns:answerCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Call-Back-Call-Related">
  <input message="tns:callBack"/>
  <output message="tns:callBackResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Call-Back-Message-Call-Related">
  <input message="tns:callBackMessage"/>
  <output message="tns:callBackMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Camp-On-Call">
  <input message="tns:camponCall"/>
  <output message="tns:camponCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Call">
  <input message="tns:clearCall"/>
  <output message="tns:clearCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Conference-Call">
```

```
<input message="tns:conferenceCall"/>
<output message="tns:conferenceCallResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Consultation-Call">
  <input message="tns:consultationCall"/>
  <output message="tns:consultationCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Dial-Digits-Call">
  <input message="tns:dialDigits"/>
  <output message="tns:dialDigitsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Directed-Pickup-Call">
  <input message="tns:directedPickupCall"/>
  <output message="tns:directedPickupCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Group-Pickup-Call">
  <input message="tns:groupPickupCall"/>
  <output message="tns:groupPickupCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Intrude-Call">
  <input message="tns:intrudeCall"/>
  <output message="tns:intrudeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Join-Call">
  <input message="tns:joinCall"/>
  <output message="tns:joinCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

```
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Predictive-Call">
  <input message="tns:makePredictiveCall"/>
  <output message="tns:makePredictiveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Park-Call">
  <input message="tns:parkCall"/>
  <output message="tns:parkCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reconnect-Call">
  <input message="tns:reconnectCall"/>
  <output message="tns:reconnectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-Message">
  <input message="tns:sendMessage"/>
  <output message="tns:sendMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Conference-Call">
  <input message="tns:singleStepConferenceCall"/>
  <output message="tns:singleStepConferenceCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Transfer-Call">
  <input message="tns:transferCall"/>
  <output message="tns:transferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

```
</operation>
<operation name="CSTA-Bridged-event">
  <output message="tns:bridgedEvent"/>
</operation>
<operation name="CSTA-Call-Cleared-event">
  <output message="tns:callClearedEvent"/>
</operation>
<operation name="CSTA-Conferenced-event">
  <output message="tns:conferencedEvent"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Digits-Dialed-event">
  <output message="tns:digitsDialedEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Capabilities-Changed-event">
  <output message="tns:networkCapabilitiesChangedEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Offered-event">
  <output message="tns:offeredEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Queued-event">
```

```
<output message="tns:queuedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
```



## 12 Call Associated Features - Messages and Operations

This clause specifies the WSDL messages and operations for the call associated features.

### 12.1 Messages

#### message **associateData**

```
parts    parameter
         element  csta:AssociateData
source  <message name="associateData">
         <part name="parameter" element="csta:AssociateData"/>
         </message>
```

#### message **associateDataResponse**

```
parts    parameter
         element  csta:AssociateDataResponse
source  <message name="associateDataResponse">
         <part name="parameter" element="csta:AssociateDataResponse"/>
         </message>
```

#### message **cancelTelephonyTones**

```
parts    parameter
         element  csta:CancelTelephonyTones
source  <message name="cancelTelephonyTones">
         <part name="parameter" element="csta:CancelTelephonyTones"/>
         </message>
```

#### message **cancelTelephonyTonesResponse**

```
parts    parameter
         element  csta:CancelTelephonyTonesResponse
source  <message name="cancelTelephonyTonesResponse">
         <part name="parameter" element="csta:CancelTelephonyTonesResponse"/>
         </message>
```

### message **changeConnectionInformation**

parts	<b>parameter</b>
	element <b>csta:ChangeConnectionInformation</b>
source	<code>&lt;message name="changeConnectionInformation"&gt;</code> <code>&lt;part name="parameter" element="csta:ChangeConnectionInformation"/&gt;</code> <code>&lt;/message&gt;</code>

### message **changeConnectionInformationResponse**

parts	<b>parameter</b>
	element <b>csta:ChangeConnectionInformationResponse</b>
source	<code>&lt;message name="changeConnectionInformationResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ChangeConnectionInformationResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **generateDigits**

parts	<b>parameter</b>
	element <b>csta:GenerateDigits</b>
source	<code>&lt;message name="generateDigits"&gt;</code> <code>&lt;part name="parameter" element="csta:GenerateDigits"/&gt;</code> <code>&lt;/message&gt;</code>

### message **generateDigitsResponse**

parts	<b>parameter</b>
	element <b>csta:GenerateDigitsResponse</b>
source	<code>&lt;message name="generateDigitsResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:GenerateDigitsResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **generateTelephonyTones**

parts	<b>parameter</b>
	element <b>csta:GenerateTelephonyTones</b>
source	<code>&lt;message name="generateTelephonyTones"&gt;</code> <code>&lt;part name="parameter" element="csta:GenerateTelephonyTones"/&gt;</code> <code>&lt;/message&gt;</code>

### message **generateTelephonyTonesResponse**

```
parts      parameter
           element  csta:GenerateTelephonyTonesResponse
source    <message name="generateTelephonyTonesResponse">
           <part name="parameter" element="csta:GenerateTelephonyTonesResponse"/>
           </message>
```

### message **sendUserInformation**

```
parts      parameter
           element  csta:SendUserInformation
source    <message name="sendUserInformation">
           <part name="parameter" element="csta:SendUserInformation"/>
           </message>
```

### message **sendUserInformationResponse**

```
parts      parameter
           element  csta:SendUserInformationResponse
source    <message name="sendUserInformationResponse">
           <part name="parameter" element="csta:SendUserInformationResponse"/>
           </message>
```

### message **callInformationEvent**

```
parts      parameter
           element  csta:CallInformationEvent
source    <message name="callInformationEvent">
           <part name="parameter" element="csta:CallInformationEvent"/>
           </message>
```

### message **chargingEvent**

```
parts      parameter
           element  csta:ChargingEvent
source    <message name="chargingEvent">
           <part name="parameter" element="csta:ChargingEvent"/>
           </message>
```

### message **digitsGeneratedEvent**

```
parts      parameter
           element  csta:DigitsGeneratedEvent
source    <message name="digitsGeneratedEvent">
           <part name="parameter" element="csta:DigitsGeneratedEvent"/>
           </message>
```

### message **telephonyTonesGeneratedEvent**

```
parts      parameter
           element  csta:TelephonyTonesGeneratedEvent
source    <message name="telephonyTonesGeneratedEvent">
           <part name="parameter" element="csta:TelephonyTonesGeneratedEvent"/>
           </message>
```

### message **serviceCompletionFailureEvent**

```
parts      parameter
           element  csta:ServiceCompletionFailureEvent
source    <message name="serviceCompletionFailureEvent">
           <part name="parameter" element="csta:ServiceCompletionFailureEvent"/>
           </message>
```



## 12.2 Operations

operations	<b>CSTA-Associate-Data</b>
	input <a href="#"><u>tns:associateData</u></a>
	output <a href="#"><u>tns:associateDataResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Cancel-Telephony-Tones</b>
	input <a href="#"><u>tns:cancelTelephonyTones</u></a>
	output <a href="#"><u>tns:cancelTelephonyTonesResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Change-Connection-Information</b>
	input <a href="#"><u>tns:changeConnectionInformation</u></a>
	output <a href="#"><u>tns:changeConnectionInformationResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Generate-Digits</b>
input <a href="#"><u>tns:generateDigits</u></a>	
output <a href="#"><u>tns:generateDigitsResponse</u></a>	
fault <a href="#"><u>tns:negResponse</u></a>	
<b>CSTA-Generate-Telephony-Tones</b>	
input <a href="#"><u>tns:generateTelephonyTones</u></a>	
output <a href="#"><u>tns:generateTelephonyTonesResponse</u></a>	
fault <a href="#"><u>tns:negResponse</u></a>	
<b>CSTA-Send-User-Information</b>	
input <a href="#"><u>tns:sendUserInformation</u></a>	
output <a href="#"><u>tns:sendUserInformationResponse</u></a>	
fault <a href="#"><u>tns:negResponse</u></a>	
<b>CSTA-Call-Information-event</b>	
output <a href="#"><u>tns:callInformationEvent</u></a>	
<b>CSTA-Charging-event</b>	
output <a href="#"><u>tns:chargingEvent</u></a>	
<b>CSTA-Digits-Generated-event</b>	
output <a href="#"><u>tns:digitsGeneratedEvent</u></a>	
<b>CSTA-Telephony-Tones-Generated-event</b>	
output <a href="#"><u>tns:telephonyTonesGeneratedEvent</u></a>	
<b>CSTA-Service-Completion-Failure-event</b>	
output <a href="#"><u>tns:serviceCompletionFailureEvent</u></a>	

```
source <operation name="CSTA-Associate-Data">
  <input message="tns:associateData"/>
  <output message="tns:associateDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Cancel-Telephony-Tones">
  <input message="tns:cancelTelephonyTones"/>
  <output message="tns:cancelTelephonyTonesResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Change-Connection-Information">
  <input message="tns:changeConnectionInformation"/>
  <output message="tns:changeConnectionInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Generate-Digits">
  <input message="tns:generateDigits"/>
  <output message="tns:generateDigitsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Generate-Telephony-Tones">
  <input message="tns:generateTelephonyTones"/>
  <output message="tns:generateTelephonyTonesResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-User-Information">
  <input message="tns:sendUserInformation"/>
  <output message="tns:sendUserInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Call-Information-event">
  <output message="tns:callInformationEvent"/>
</operation>
<operation name="CSTA-Charging-event">
  <output message="tns:chargingEvent"/>
</operation>
<operation name="CSTA-Digits-Generated-event">
  <output message="tns:digitsGeneratedEvent"/>
</operation>
<operation name="CSTA-Telephony-Tones-Generated-event">
  <output message="tns:telephonyTonesGeneratedEvent"/>
</operation>
<operation name="CSTA-Service-Completion-Failure-event">
  <output message="tns:serviceCompletionFailureEvent"/>
</operation>
```

## 13 Media Attachment Features - Messages and Operations

This clause specifies the WSDL messages and operations for the media attachment features.

### 13.1 Messages

#### message **attachMediaService**

```
parts    parameter
         element  csta:AttachMediaService
source   <message name="attachMediaService">
         <part name="parameter" element="csta:AttachMediaService"/>
         </message>
```

#### message **attachMediaServiceResponse**

```
parts    parameter
         element  csta:AttachMediaServiceResponse
source   <message name="attachMediaServiceResponse">
         <part name="parameter" element="csta:AttachMediaServiceResponse"/>
         </message>
```

#### message **detachMediaService**

```
parts    parameter
         element  csta:DetachMediaService
source   <message name="detachMediaService">
         <part name="parameter" element="csta:DetachMediaService"/>
         </message>
```

#### message **detachMediaServiceResponse**

```
parts    parameter
         element  csta:DetachMediaServiceResponse
source   <message name="detachMediaServiceResponse">
         <part name="parameter" element="csta:DetachMediaServiceResponse"/>
         </message>
```

### message **mediaAttachedEvent**

```
parts    parameter
        element csta:MediaAttachedEvent
source  <message name="mediaAttachedEvent">
        <part name="parameter" element="csta:MediaAttachedEvent"/>
        </message>
```

### message **mediaDetachedEvent**

```
parts    parameter
        element csta:MediaDetachedEvent
source  <message name="mediaDetachedEvent">
        <part name="parameter" element="csta:MediaDetachedEvent"/>
        </message>
```



## 13.2 Operations

operations	<p><b>CSTA-Attach-Media-Service</b></p> <p>input <a href="#">tns:attachMediaService</a></p> <p>output <a href="#">tns:attachMediaServiceResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Detach-Media-Service</b></p> <p>input <a href="#">tns:detachMediaService</a></p> <p>output <a href="#">tns:detachMediaServiceResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Media-Attached-Event</b></p> <p>output <a href="#">tns:mediaAttachedEvent</a></p> <p><b>CSTA-Media-Detached-Event</b></p> <p>output <a href="#">tns:mediaDetachedEvent</a></p>
source	<pre> &lt;operation name="CSTA-Attach-Media-Service"&gt;   &lt;input message="tns:attachMediaService"/&gt;   &lt;output message="tns:attachMediaServiceResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Detach-Media-Service"&gt;   &lt;input message="tns:detachMediaService"/&gt;   &lt;output message="tns:detachMediaServiceResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Media-Attached-Event"&gt;   &lt;output message="tns:mediaAttachedEvent"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Media-Detached-Event"&gt;   &lt;output message="tns:mediaDetachedEvent"/&gt; &lt;/operation&gt; </pre>

## 14 Routing Features - Messages and Operations

This clause specifies the WSDL messages and operations for the routing features.

### 14.1 Messages

#### message **routeRegister**

```
parts    parameter
         element    csta:RouteRegister
source   <message name="routeRegister">
         <part name="parameter" element="csta:RouteRegister"/>
         </message>
```

#### message **routeRegisterResponse**

```
parts    parameter
         element    csta:RouteRegisterResponse
source   <message name="routeRegisterResponse">
         <part name="parameter" element="csta:RouteRegisterResponse"/>
         </message>
```

#### message **routeRegisterAbort**

```
parts    parameter
         element    csta:RouteRegisterAbort
source   <message name="routeRegisterAbort">
         <part name="parameter" element="csta:RouteRegisterAbort"/>
         </message>
```

#### message **routeRegisterCancel**

```
parts    parameter
         element    csta:RouteRegisterCancel
source   <message name="routeRegisterCancel">
         <part name="parameter" element="csta:RouteRegisterCancel"/>
         </message>
```

### message routeRegisterCancelResponse

parts	<b>parameter</b>
	element <b>csta:RouteRegisterCancelResponse</b>
source	<code>&lt;message name="routeRegisterCancelResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteRegisterCancelResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message reRoute

parts	<b>parameter</b>
	element <b>csta:ReRoute</b>
source	<code>&lt;message name="reRoute"&gt;</code> <code>&lt;part name="parameter" element="csta:ReRoute"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeEnd

parts	<b>parameter</b>
	element <b>csta:RouteEnd</b>
source	<code>&lt;message name="routeEnd"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteEnd"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeReject

parts	<b>parameter</b>
	element <b>csta:RouteReject</b>
source	<code>&lt;message name="routeReject"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteReject"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeRequest

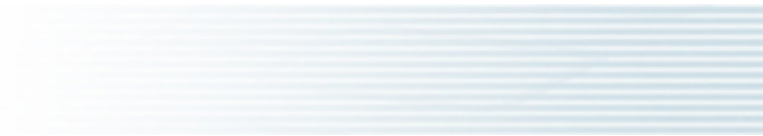
parts	<b>parameter</b>
	element <b>csta:RouteRequest</b>
source	<code>&lt;message name="routeRequest"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteRequest"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeSelect

parts	<b>parameter</b>
	element <b>csta:RouteSelect</b>
source	<code>&lt;message name="routeSelect"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteSelect"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeUsed

parts	<b>parameter</b>
	element <b>csta:RouteUsed</b>
source	<code>&lt;message name="routeUsed"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteUsed"/&gt;</code> <code>&lt;/message&gt;</code>



## 14.2 Operations

operations

### CSTA-Route-Register

input [tns:routeRegister](#)  
 output [tns:routeRegisterResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Route-Register-Abort

output [tns:routeRegisterAbort](#)

### CSTA-Route-Register-Cancel

input [tns:routeRegisterCancel](#)  
 output [tns:routeRegisterCancelResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Re-Route

output [tns:reRoute](#)

### CSTA-Route-End-as-input

input [tns:routeEnd](#)

### CSTA-Route-End-as-output

output [tns:routeEnd](#)

### CSTA-Route-Reject

input [tns:routeReject](#)

### CSTA-Route-Request

output [tns:routeRequest](#)

### CSTA-Route-Select

input [tns:routeSelect](#)

### CSTA-Route-Used

output [tns:routeUsed](#)

source

```
<operation name="CSTA-Route-Register">
  <input message="tns:routeRegister"/>
  <output message="tns:routeRegisterResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Route-Register-Abort">
  <output message="tns:routeRegisterAbort"/>
</operation>
<operation name="CSTA-Route-Register-Cancel">
```

```
<input message="tns:routeRegisterCancel"/>
<output message="tns:routeRegisterCancelResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Re-Route">
  <output message="tns:reRoute"/>
</operation>
<operation name="CSTA-Route-End-as-input">
  <input message="tns:routeEnd"/>
</operation>
<operation name="CSTA-Route-End-as-output">
  <output message="tns:routeEnd"/>
</operation>
<operation name="CSTA-Route-Reject">
  <input message="tns:routeReject"/>
</operation>
<operation name="CSTA-Route-Request">
  <output message="tns:routeRequest"/>
</operation>
<operation name="CSTA-Route-Select">
  <input message="tns:routeSelect"/>
</operation>
<operation name="CSTA-Route-Used">
  <output message="tns:routeUsed"/>
</operation>
```

## 15 Physical Device Features - Messages and Operations

This clause specifies the WSDL messages and operations for the physical device features.

### 15.1 Messages

#### message **buttonPress**

```
parts    parameter
         element    csta:ButtonPress
source   <message name="buttonPress">
         <part name="parameter" element="csta:ButtonPress"/>
         </message>
```

#### message **buttonPressResponse**

```
parts    parameter
         element    csta:ButtonPressResponse
source   <message name="buttonPressResponse">
         <part name="parameter" element="csta:ButtonPressResponse"/>
         </message>
```

#### message **getAuditoryApparatusInformation**

```
parts    parameter
         element    csta:GetAuditoryApparatusInformation
source   <message name="getAuditoryApparatusInformation">
         <part name="parameter" element="csta:GetAuditoryApparatusInformation"/>
         </message>
```

#### message **getAuditoryApparatusInformationResponse**

```
parts    parameter
         element    csta:GetAuditoryApparatusInformationResponse
source   <message name="getAuditoryApparatusInformationResponse">
         <part name="parameter" element="csta:GetAuditoryApparatusInformationResponse"/>
         </message>
```

### message **getButtonInformation**

```
parts    parameter
         element    csta:GetButtonInformation
source   <message name="getButtonInformation">
         <part name="parameter" element="csta:GetButtonInformation"/>
         </message>
```

### message **getButtonInformationResponse**

```
parts    parameter
         element    csta:GetButtonInformationResponse
source   <message name="getButtonInformationResponse">
         <part name="parameter" element="csta:GetButtonInformationResponse"/>
         </message>
```

### message **getDisplay**

```
parts    parameter
         element    csta:GetDisplay
source   <message name="getDisplay">
         <part name="parameter" element="csta:GetDisplay"/>
         </message>
```

### message **getDisplayResponse**

```
parts    parameter
         element    csta:GetDisplayResponse
source   <message name="getDisplayResponse">
         <part name="parameter" element="csta:GetDisplayResponse"/>
         </message>
```

### message **getHookswitchStatus**

```
parts    parameter
         element    csta:GetHookswitchStatus
source   <message name="getHookswitchStatus">
         <part name="parameter" element="csta:GetHookswitchStatus"/>
         </message>
```



### message **getHookswitchStatusResponse**

```
parts      parameter
           element  csta:GetHookswitchStatusResponse
source     <message name="getHookswitchStatusResponse">
           <part name="parameter" element="csta:GetHookswitchStatusResponse"/>
           </message>
```

### message **getLampInformation**

```
parts      parameter
           element  csta:GetLampInformation
source     <message name="getLampInformation">
           <part name="parameter" element="csta:GetLampInformation"/>
           </message>
```

### message **getLampInformationResponse**

```
parts      parameter
           element  csta:GetLampInformationResponse
source     <message name="getLampInformationResponse">
           <part name="parameter" element="csta:GetLampInformationResponse"/>
           </message>
```

### message **getLampMode**

```
parts      parameter
           element  csta:GetLampMode
source     <message name="getLampMode">
           <part name="parameter" element="csta:GetLampMode"/>
           </message>
```

### message **getLampModeResponse**

parts	<b>parameter</b>
	element <b>csta:GetLampModeResponse</b>
source	<code>&lt;message name="getLampModeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:GetLampModeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getMessageWaitingIndicator**

parts	<b>parameter</b>
	element <b>csta:GetMessageWaitingIndicator</b>
source	<code>&lt;message name="getMessageWaitingIndicator"&gt;</code> <code>&lt;part name="parameter" element="csta:GetMessageWaitingIndicator"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getMessageWaitingIndicatorResponse**

parts	<b>parameter</b>
	element <b>csta:GetMessageWaitingIndicatorResponse</b>
source	<code>&lt;message name="getMessageWaitingIndicatorResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:GetMessageWaitingIndicatorResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getMicrophoneGain**

parts	<b>parameter</b>
	element <b>csta:GetMicrophoneGain</b>
source	<code>&lt;message name="getMicrophoneGain"&gt;</code> <code>&lt;part name="parameter" element="csta:GetMicrophoneGain"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getMicrophoneGainResponse**

parts	<b>parameter</b>
	element <b>csta:GetMicrophoneGainResponse</b>
source	<code>&lt;message name="getMicrophoneGainResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:GetMicrophoneGainResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getMicrophoneMute**

```
parts    parameter
         element  csta:GetMicrophoneMute
source   <message name="getMicrophoneMute">
         <part name="parameter" element="csta:GetMicrophoneMute"/>
         </message>
```

### message **getMicrophoneMuteResponse**

```
parts    parameter
         element  csta:GetMicrophoneMuteResponse
source   <message name="getMicrophoneMuteResponse">
         <part name="parameter" element="csta:GetMicrophoneMuteResponse"/>
         </message>
```

### message **getRingerStatus**

```
parts    parameter
         element  csta:GetRingerStatus
source   <message name="getRingerStatus">
         <part name="parameter" element="csta:GetRingerStatus"/>
         </message>
```

### message **getRingerStatusResponse**

```
parts    parameter
         element  csta:GetRingerStatusResponse
source   <message name="getRingerStatusResponse">
         <part name="parameter" element="csta:GetRingerStatusResponse"/>
         </message>
```

### message **getSpeakerMute**

parts	<b>parameter</b>
	element <b>csta:GetSpeakerMute</b>
source	<pre>&lt;message name="getSpeakerMute"&gt;   &lt;part name="parameter" element="csta:GetSpeakerMute"/&gt; &lt;/message&gt;</pre>

### message **getSpeakerMuteResponse**

parts	<b>parameter</b>
	element <b>csta:GetSpeakerMuteResponse</b>
source	<pre>&lt;message name="getSpeakerMuteResponse"&gt;   &lt;part name="parameter" element="csta:GetSpeakerMuteResponse"/&gt; &lt;/message&gt;</pre>

### message **getSpeakerVolume**

parts	<b>parameter</b>
	element <b>csta:GetSpeakerVolume</b>
source	<pre>&lt;message name="getSpeakerVolume"&gt;   &lt;part name="parameter" element="csta:GetSpeakerVolume"/&gt; &lt;/message&gt;</pre>

### message **getSpeakerVolumeResponse**

parts	<b>parameter</b>
	element <b>csta:GetSpeakerVolumeResponse</b>
source	<pre>&lt;message name="getSpeakerVolumeResponse"&gt;   &lt;part name="parameter" element="csta:GetSpeakerVolumeResponse"/&gt; &lt;/message&gt;</pre>

### message **setButtonInformation**

parts	<b>parameter</b>
	element <b>csta:SetButtonInformation</b>
source	<pre>&lt;message name="setButtonInformation"&gt;   &lt;part name="parameter" element="csta:SetButtonInformation"/&gt; &lt;/message&gt;</pre>

### message **setButtonInformationResponse**

```
parts      parameter
           element  csta:SetButtonInformationResponse
source    <message name="setButtonInformationResponse">
           <part name="parameter" element="csta:SetButtonInformationResponse"/>
           </message>
```

### message **setDisplay**

```
parts      parameter
           element  csta:SetDisplay
source    <message name="setDisplay">
           <part name="parameter" element="csta:SetDisplay"/>
           </message>
```

### message **setDisplayResponse**

```
parts      parameter
           element  csta:SetDisplayResponse
source    <message name="setDisplayResponse">
           <part name="parameter" element="csta:SetDisplayResponse"/>
           </message>
```

### message **setHookswitchStatus**

```
parts      parameter
           element  csta:SetHookswitchStatus
source    <message name="setHookswitchStatus">
           <part name="parameter" element="csta:SetHookswitchStatus"/>
           </message>
```

### message **setHookswitchStatusResponse**

parts	<b>parameter</b>
	element <b>csta:SetHookswitchStatusResponse</b>
source	<code>&lt;message name="setHookswitchStatusResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetHookswitchStatusResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setLampMode**

parts	<b>parameter</b>
	element <b>csta:SetLampMode</b>
source	<code>&lt;message name="setLampMode"&gt;</code> <code>&lt;part name="parameter" element="csta:SetLampMode"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setLampModeResponse**

parts	<b>parameter</b>
	element <b>csta:SetLampModeResponse</b>
source	<code>&lt;message name="setLampModeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetLampModeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMessageWaitingIndicator**

parts	<b>parameter</b>
	element <b>csta:SetMessageWaitingIndicator</b>
source	<code>&lt;message name="setMessageWaitingIndicator"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMessageWaitingIndicator"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMessageWaitingIndicatorResponse**

parts	<b>parameter</b>
	element <b>csta:SetMessageWaitingIndicatorResponse</b>
source	<code>&lt;message name="setMessageWaitingIndicatorResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMessageWaitingIndicatorResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMicrophoneGain**

parts	<b>parameter</b>
	element <b>csta:SetMicrophoneGain</b>
source	<code>&lt;message name="setMicrophoneGain"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMicrophoneGain"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMicrophoneGainResponse**

parts	<b>parameter</b>
	element <b>csta:SetMicrophoneGainResponse</b>
source	<code>&lt;message name="setMicrophoneGainResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMicrophoneGainResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMicrophoneMute**

parts	<b>parameter</b>
	element <b>csta:SetMicrophoneMute</b>
source	<code>&lt;message name="setMicrophoneMute"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMicrophoneMute"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setMicrophoneMuteResponse**

parts	<b>parameter</b>
	element <b>csta:SetMicrophoneMuteResponse</b>
source	<code>&lt;message name="setMicrophoneMuteResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetMicrophoneMuteResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setRingerStatus**

parts	<b>parameter</b>
	element <b>csta:SetRingerStatus</b>
source	<code>&lt;message name="setRingerStatus"&gt;</code> <code>&lt;part name="parameter" element="csta:SetRingerStatus"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setRingerStatusResponse**

parts	<b>parameter</b>
	element <b>csta:SetRingerStatusResponse</b>
source	<code>&lt;message name="setRingerStatusResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetRingerStatusResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setSpeakerMute**

parts	<b>parameter</b>
	element <b>csta:SetSpeakerMute</b>
source	<code>&lt;message name="setSpeakerMute"&gt;</code> <code>&lt;part name="parameter" element="csta:SetSpeakerMute"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setSpeakerMuteResponse**

parts	<b>parameter</b>
	element <b>csta:SetSpeakerMuteResponse</b>
source	<code>&lt;message name="setSpeakerMuteResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetSpeakerMuteResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setSpeakerVolume**

parts	<b>parameter</b>
	element <b>csta:SetSpeakerVolume</b>
source	<code>&lt;message name="setSpeakerVolume"&gt;</code> <code>&lt;part name="parameter" element="csta:SetSpeakerVolume"/&gt;</code> <code>&lt;/message&gt;</code>



### message **setSpeakerVolumeResponse**

```
parts      parameter
           element  csta:SetSpeakerVolumeResponse
source    <message name="setSpeakerVolumeResponse">
           <part name="parameter" element="csta:SetSpeakerVolumeResponse"/>
           </message>
```

### message **buttonInformationEvent**

```
parts      parameter
           element  csta:ButtonInformationEvent
source    <message name="buttonInformationEvent">
           <part name="parameter" element="csta:ButtonInformationEvent"/>
           </message>
```

### message **buttonPressEvent**

```
parts      parameter
           element  csta:ButtonPressEvent
source    <message name="buttonPressEvent">
           <part name="parameter" element="csta:ButtonPressEvent"/>
           </message>
```

### message **displayUpdatedEvent**

```
parts      parameter
           element  csta:DisplayUpdatedEvent
source    <message name="displayUpdatedEvent">
           <part name="parameter" element="csta:DisplayUpdatedEvent"/>
           </message>
```

### message hookswitchEvent

```
parts    parameter
         element    csta:HookswitchEvent
source   <message name="hookswitchEvent">
         <part name="parameter" element="csta:HookswitchEvent"/>
         </message>
```

### message lampModeEvent

```
parts    parameter
         element    csta:LampModeEvent
source   <message name="lampModeEvent">
         <part name="parameter" element="csta:LampModeEvent"/>
         </message>
```

### message messageWaitingEvent

```
parts    parameter
         element    csta:MessageWaitingEvent
source   <message name="messageWaitingEvent">
         <part name="parameter" element="csta:MessageWaitingEvent"/>
         </message>
```

### message microphoneGainEvent

```
parts    parameter
         element    csta:MicrophoneGainEvent
source   <message name="microphoneGainEvent">
         <part name="parameter" element="csta:MicrophoneGainEvent"/>
         </message>
```

### message microphoneMuteEvent

```
parts    parameter
         element    csta:MicrophoneMuteEvent
source   <message name="microphoneMuteEvent">
         <part name="parameter" element="csta:MicrophoneMuteEvent"/>
         </message>
```

### message **ringerStatusEvent**

```
parts    parameter
         element csta:RingerStatusEvent
source  <message name="ringerStatusEvent">
         <part name="parameter" element="csta:RingerStatusEvent"/>
       </message>
```

### message **speakerMuteEvent**

```
parts    parameter
         element csta:SpeakerMuteEvent
source  <message name="speakerMuteEvent">
         <part name="parameter" element="csta:SpeakerMuteEvent"/>
       </message>
```

### message **speakerVolumeEvent**

```
parts    parameter
         element csta:SpeakerVolumeEvent
source  <message name="speakerVolumeEvent">
         <part name="parameter" element="csta:SpeakerVolumeEvent"/>
       </message>
```

## 15.2 Operations

operations

### CSTA-Button-Press

input [tns:buttonPress](#)  
output [tns:buttonPressResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Auditory-Apparatus-Information

input [tns:getAuditoryApparatusInformation](#)  
output [tns:getAuditoryApparatusInformationResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Button-Information

input [tns:getButtonInformation](#)  
output [tns:getButtonInformationResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Display

input [tns:getDisplay](#)  
output [tns:getDisplayResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Hookswitch-Status

input [tns:getHookswitchStatus](#)  
output [tns:getHookswitchStatusResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Lamp-Information

input [tns:getLampInformation](#)  
output [tns:getLampInformationResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Lamp-Mode

input [tns:getLampMode](#)  
output [tns:getLampModeResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Message-Waiting-Indicator

input [tns:getMessageWaitingIndicator](#)  
output [tns:getMessageWaitingIndicatorResponse](#)  
fault [tns:neqResponse](#)

### CSTA-Get-Microphone-Gain

input [tns:getMicrophoneGain](#)  
output [tns:getMicrophoneGainResponse](#)  
fault [tns:neqResponse](#)

**CSTA-Get-Microphone-Mute**

input [tns:getMicrophoneMute](#)  
output [tns:getMicrophoneMuteResponse](#)  
fault [tns:negResponse](#)

**CSTA-Get-Ringer-Status**

input [tns:getRingerStatus](#)  
output [tns:getRingerStatusResponse](#)  
fault [tns:negResponse](#)

**CSTA-Get-Speaker-Mute**

input [tns:getSpeakerMute](#)  
output [tns:getSpeakerMuteResponse](#)  
fault [tns:negResponse](#)

**CSTA-Get-Speaker-Volume**

input [tns:getSpeakerVolume](#)  
output [tns:getSpeakerVolumeResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Button-Information**

input [tns:setButtonInformation](#)  
output [tns:setButtonInformationResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Display**

input [tns:setDisplay](#)  
output [tns:setDisplayResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Hookswitch-Status**

input [tns:setHookswitchStatus](#)  
output [tns:setHookswitchStatusResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Lamp-Mode**

input [tns:setLampMode](#)  
output [tns:setLampModeResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Message-Waiting-Indicator**

input [tns:setMessageWaitingIndicator](#)  
output [tns:setMessageWaitingIndicatorResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Microphone-Gain**

input [tns:setMicrophoneGain](#)  
output [tns:setMicrophoneGainResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Microphone-Mute**

input [tns:setMicrophoneMute](#)  
output [tns:setMicrophoneMuteResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Ringer-Status**

input [tns:setRingerStatus](#)  
output [tns:setRingerStatusResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Speaker-Mute**

input [tns:setSpeakerMute](#)  
output [tns:setSpeakerMuteResponse](#)  
fault [tns:negResponse](#)

**CSTA-Set-Speaker-Volume**

input [tns:setSpeakerVolume](#)  
output [tns:setSpeakerVolumeResponse](#)  
fault [tns:negResponse](#)

**CSTA-Button-Information-Event**

output [tns:buttonInformationEvent](#)

**CSTA-Button-Press-Event**

output [tns:buttonPressEvent](#)

**CSTA-Display-Updated-Event**

output [tns:displayUpdatedEvent](#)

**CSTA-Hookswitch-Event**

output [tns:hookswitchEvent](#)

**CSTA-Lamp-Mode-Event**

output [tns:lampModeEvent](#)

**CSTA-Message-Waiting-Event**

output [tns:messageWaitingEvent](#)

**CSTA-Microphone-Gain-Event**

output [tns:microphoneGainEvent](#)

#### CSTA-Microphone-Mute-Event

output [tns:microphoneMuteEvent](#)

#### CSTA-Ringer-Status-Event

output [tns:ringerStatusEvent](#)

#### CSTA-Speaker-Mute-Event

output [tns:speakerMuteEvent](#)

#### CSTA-Speaker-Volume-Event

output [tns:speakerVolumeEvent](#)

source

```

<operation name="CSTA-Button-Press">
  <input message="tns:buttonPress"/>
  <output message="tns:buttonPressResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Auditory-Apparatus-Information">
  <input message="tns:getAuditoryApparatusInformation"/>
  <output message="tns:getAuditoryApparatusInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Button-Information">
  <input message="tns:getButtonInformation"/>
  <output message="tns:getButtonInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Display">
  <input message="tns:getDisplay"/>
  <output message="tns:getDisplayResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Hookswitch-Status">
  <input message="tns:getHookswitchStatus"/>
  <output message="tns:getHookswitchStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Lamp-Information">
  <input message="tns:getLampInformation"/>
  <output message="tns:getLampInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Lamp-Mode">
  <input message="tns:getLampMode"/>
  <output message="tns:getLampModeResponse"/>

```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Message-Waiting-Indicator">
  <input message="tns:getMessageWaitingIndicator"/>
  <output message="tns:getMessageWaitingIndicatorResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Microphone-Gain">
  <input message="tns:getMicrophoneGain"/>
  <output message="tns:getMicrophoneGainResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Microphone-Mute">
  <input message="tns:getMicrophoneMute"/>
  <output message="tns:getMicrophoneMuteResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Ringer-Status">
  <input message="tns:getRingerStatus"/>
  <output message="tns:getRingerStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Speaker-Mute">
  <input message="tns:getSpeakerMute"/>
  <output message="tns:getSpeakerMuteResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Speaker-Volume">
  <input message="tns:getSpeakerVolume"/>
  <output message="tns:getSpeakerVolumeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Button-Information">
  <input message="tns:setButtonInformation"/>
  <output message="tns:setButtonInformationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Display">
  <input message="tns:setDisplay"/>
  <output message="tns:setDisplayResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Hookswitch-Status">
  <input message="tns:setHookswitchStatus"/>
```



```
<output message="tns:setHookswitchStatusResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Lamp-Mode">
  <input message="tns:setLampMode"/>
  <output message="tns:setLampModeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Message-Waiting-Indicator">
  <input message="tns:setMessageWaitingIndicator"/>
  <output message="tns:setMessageWaitingIndicatorResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Microphone-Gain">
  <input message="tns:setMicrophoneGain"/>
  <output message="tns:setMicrophoneGainResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Microphone-Mute">
  <input message="tns:setMicrophoneMute"/>
  <output message="tns:setMicrophoneMuteResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Ringer-Status">
  <input message="tns:setRingerStatus"/>
  <output message="tns:setRingerStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Speaker-Mute">
  <input message="tns:setSpeakerMute"/>
  <output message="tns:setSpeakerMuteResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Speaker-Volume">
  <input message="tns:setSpeakerVolume"/>
  <output message="tns:setSpeakerVolumeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Button-Information-Event">
  <output message="tns:buttonInformationEvent"/>
</operation>
<operation name="CSTA-Button-Press-Event">
  <output message="tns:buttonPressEvent"/>
</operation>
```

```
<operation name="CSTA-Display-Updated-Event">
  <output message="tns:displayUpdatedEvent"/>
</operation>
<operation name="CSTA-Hookswitch-Event">
  <output message="tns:hookswitchEvent"/>
</operation>
<operation name="CSTA-Lamp-Mode-Event">
  <output message="tns:lampModeEvent"/>
</operation>
<operation name="CSTA-Message-Waiting-Event">
  <output message="tns:messageWaitingEvent"/>
</operation>
<operation name="CSTA-Microphone-Gain-Event">
  <output message="tns:microphoneGainEvent"/>
</operation>
<operation name="CSTA-Microphone-Mute-Event">
  <output message="tns:microphoneMuteEvent"/>
</operation>
<operation name="CSTA-Ringer-Status-Event">
  <output message="tns:ringerStatusEvent"/>
</operation>
<operation name="CSTA-Speaker-Mute-Event">
  <output message="tns:speakerMuteEvent"/>
</operation>
<operation name="CSTA-Speaker-Volume-Event">
  <output message="tns:speakerVolumeEvent"/>
</operation>
```

## 16 Logical Device Features - Messages and Operations

This clause specifies the WSDL messages and operations for the logical device features.

### 16.1 Messages

#### message **callBackMessageNonCallRelated**

```
parts    parameter
         element  csta:CallBackMessageNonCallRelated
source  <message name="callBackMessageNonCallRelated">
         <part name="parameter" element="csta:CallBackMessageNonCallRelated"/>
       </message>
```

#### message **callBackMessageNonCallRelatedResponse**

```
parts    parameter
         element  csta:CallBackMessageNonCallRelatedResponse
source  <message name="callBackMessageNonCallRelatedResponse">
         <part name="parameter" element="csta:CallBackMessageNonCallRelatedResponse"/>
       </message>
```

#### message **callBackNonCallRelated**

```
parts    parameter
         element  csta:CallBackNonCallRelated
source  <message name="callBackNonCallRelated">
         <part name="parameter" element="csta:CallBackNonCallRelated"/>
       </message>
```

#### message **callBackNonCallRelatedResponse**

```
parts    parameter
         element  csta:CallBackNonCallRelatedResponse
source  <message name="callBackNonCallRelatedResponse">
         <part name="parameter" element="csta:CallBackNonCallRelatedResponse"/>
       </message>
```

### message **cancelCallBack**

parts	<b>parameter</b>
	element <b>csta:CancelCallBack</b>
source	<code>&lt;message name="cancelCallBack"&gt;</code> <code>&lt;part name="parameter" element="csta:CancelCallBack"/&gt;</code> <code>&lt;/message&gt;</code>

### message **cancelCallBackResponse**

parts	<b>parameter</b>
	element <b>csta:CancelCallBackResponse</b>
source	<code>&lt;message name="cancelCallBackResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:CancelCallBackResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **cancelCallBackMessage**

parts	<b>parameter</b>
	element <b>csta:CancelCallBackMessage</b>
source	<code>&lt;message name="cancelCallBackMessage"&gt;</code> <code>&lt;part name="parameter" element="csta:CancelCallBackMessage"/&gt;</code> <code>&lt;/message&gt;</code>

### message **cancelCallBackMessageResponse**

parts	<b>parameter</b>
	element <b>csta:CancelCallBackMessageResponse</b>
source	<code>&lt;message name="cancelCallBackMessageResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:CancelCallBackMessageResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getAgentState**

parts	<b>parameter</b>
	element <b>csta:GetAgentState</b>
source	<code>&lt;message name="getAgentState"&gt;</code> <code>&lt;part name="parameter" element="csta:GetAgentState"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getAgentStateResponse**

```
parts    parameter
         element  csta:GetAgentStateResponse
source  <message name="getAgentStateResponse">
         <part name="parameter" element="csta:GetAgentStateResponse"/>
         </message>
```

### message **getAutoAnswer**

```
parts    parameter
         element  csta:GetAutoAnswer
source  <message name="getAutoAnswer">
         <part name="parameter" element="csta:GetAutoAnswer"/>
         </message>
```

### message **getAutoAnswerResponse**

```
parts    parameter
         element  csta:GetAutoAnswerResponse
source  <message name="getAutoAnswerResponse">
         <part name="parameter" element="csta:GetAutoAnswerResponse"/>
         </message>
```

### message **getAutoWorkMode**

```
parts    parameter
         element  csta:GetAutoWorkMode
source  <message name="getAutoWorkMode">
         <part name="parameter" element="csta:GetAutoWorkMode"/>
         </message>
```

### message **getAutoWorkModeResponse**

parts	<b>parameter</b>
	element <b>csta:GetAutoWorkModeResponse</b>
source	<pre>&lt;message name="getAutoWorkModeResponse"&gt;   &lt;part name="parameter" element="csta:GetAutoWorkModeResponse"/&gt; &lt;/message&gt;</pre>

### message **getCallerIDStatus**

parts	<b>parameter</b>
	element <b>csta:GetCallerIDStatus</b>
source	<pre>&lt;message name="getCallerIDStatus"&gt;   &lt;part name="parameter" element="csta:GetCallerIDStatus"/&gt; &lt;/message&gt;</pre>

### message **getCallerIDStatusResponse**

parts	<b>parameter</b>
	element <b>csta:GetCallerIDStatusResponse</b>
source	<pre>&lt;message name="getCallerIDStatusResponse"&gt;   &lt;part name="parameter" element="csta:GetCallerIDStatusResponse"/&gt; &lt;/message&gt;</pre>

### message **getDoNotDisturb**

parts	<b>parameter</b>
	element <b>csta:GetDoNotDisturb</b>
source	<pre>&lt;message name="getDoNotDisturb"&gt;   &lt;part name="parameter" element="csta:GetDoNotDisturb"/&gt; &lt;/message&gt;</pre>

### message **getDoNotDisturbResponse**

parts	<b>parameter</b>
	element <b>csta:GetDoNotDisturbResponse</b>
source	<pre>&lt;message name="getDoNotDisturbResponse"&gt;   &lt;part name="parameter" element="csta:GetDoNotDisturbResponse"/&gt; &lt;/message&gt;</pre>

### message **getForwarding**

```
parts    parameter
        element csta:GetForwarding
source  <message name="getForwarding">
        <part name="parameter" element="csta:GetForwarding"/>
</message>
```

### message **getForwardingResponse**

```
parts    parameter
        element csta:GetForwardingResponse
source  <message name="getForwardingResponse">
        <part name="parameter" element="csta:GetForwardingResponse"/>
</message>
```

### message **getLastNumberDialed**

```
parts    parameter
        element csta:GetLastNumberDialed
source  <message name="getLastNumberDialed">
        <part name="parameter" element="csta:GetLastNumberDialed"/>
</message>
```

### message **getLastNumberDialedResponse**

```
parts    parameter
        element csta:GetLastNumberDialedResponse
source  <message name="getLastNumberDialedResponse">
        <part name="parameter" element="csta:GetLastNumberDialedResponse"/>
</message>
```

### message **getRouteingMode**

parts	<b>parameter</b>
	element <b>csta:GetRouteingMode</b>
source	<code>&lt;message name="getRouteingMode"&gt;</code> <code>&lt;part name="parameter" element="csta:GetRouteingMode"/&gt;</code> <code>&lt;/message&gt;</code>

### message **getRouteingModeResponse**

parts	<b>parameter</b>
	element <b>csta:GetRouteingModeResponse</b>
source	<code>&lt;message name="getRouteingModeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:GetRouteingModeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setAgentState**

parts	<b>parameter</b>
	element <b>csta:SetAgentState</b>
source	<code>&lt;message name="setAgentState"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAgentState"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setAgentStateResponse**

parts	<b>parameter</b>
	element <b>csta:SetAgentStateResponse</b>
source	<code>&lt;message name="setAgentStateResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAgentStateResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setAutoAnswer**

parts	<b>parameter</b>
	element <b>csta:SetAutoAnswer</b>
source	<code>&lt;message name="setAutoAnswer"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAutoAnswer"/&gt;</code> <code>&lt;/message&gt;</code>



### message **setAutoAnswerResponse**

parts	<b>parameter</b>
	element <b>csta:SetAutoAnswerResponse</b>
source	<code>&lt;message name="setAutoAnswerResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAutoAnswerResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setAutoWorkMode**

parts	<b>parameter</b>
	element <b>csta:SetAutoWorkMode</b>
source	<code>&lt;message name="setAutoWorkMode"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAutoWorkMode"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setAutoWorkModeResponse**

parts	<b>parameter</b>
	element <b>csta:SetAutoWorkModeResponse</b>
source	<code>&lt;message name="setAutoWorkModeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetAutoWorkModeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setCallerIDStatus**

parts	<b>parameter</b>
	element <b>csta:SetCallerIDStatus</b>
source	<code>&lt;message name="setCallerIDStatus"&gt;</code> <code>&lt;part name="parameter" element="csta:SetCallerIDStatus"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setCallerIDStatusResponse**

```
parts      parameter
           element  csta:SetCallerIDStatusResponse
source    <message name="setCallerIDStatusResponse">
           <part name="parameter" element="csta:SetCallerIDStatusResponse"/>
           </message>
```

### message **setDoNotDisturb**

```
parts      parameter
           element  csta:SetDoNotDisturb
source    <message name="setDoNotDisturb">
           <part name="parameter" element="csta:SetDoNotDisturb"/>
           </message>
```

### message **setDoNotDisturbResponse**

```
parts      parameter
           element  csta:SetDoNotDisturbResponse
source    <message name="setDoNotDisturbResponse">
           <part name="parameter" element="csta:SetDoNotDisturbResponse"/>
           </message>
```

### message **setForwarding**

```
parts      parameter
           element  csta:SetForwarding
source    <message name="setForwarding">
           <part name="parameter" element="csta:SetForwarding"/>
           </message>
```

### message **setForwardingResponse**

```
parts      parameter
           element  csta:SetForwardingResponse
source    <message name="setForwardingResponse">
           <part name="parameter" element="csta:SetForwardingResponse"/>
           </message>
```

### message **setRouteingMode**

parts	<b>parameter</b>
	element <b>csta:SetRouteingMode</b>
source	<code>&lt;message name="setRouteingMode"&gt;</code> <code>&lt;part name="parameter" element="csta:SetRouteingMode"/&gt;</code> <code>&lt;/message&gt;</code>

### message **setRouteingModeResponse**

parts	<b>parameter</b>
	element <b>csta:SetRouteingModeResponse</b>
source	<code>&lt;message name="setRouteingModeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SetRouteingModeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentBusyEvent**

parts	<b>parameter</b>
	element <b>csta:AgentBusyEvent</b>
source	<code>&lt;message name="agentBusyEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentBusyEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentLoggedOffEvent**

parts	<b>parameter</b>
	element <b>csta:AgentLoggedOffEvent</b>
source	<code>&lt;message name="agentLoggedOffEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentLoggedOffEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentLoggedInEvent**

parts	<b>parameter</b>
	element <b>csta:AgentLoggedInEvent</b>
source	<code>&lt;message name="agentLoggedInEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentLoggedInEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentNotReadyEvent**

parts	<b>parameter</b>
	element <b>csta:AgentNotReadyEvent</b>
source	<code>&lt;message name="agentNotReadyEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentNotReadyEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentReadyEvent**

parts	<b>parameter</b>
	element <b>csta:AgentReadyEvent</b>
source	<code>&lt;message name="agentReadyEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentReadyEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **agentWorkingAfterCallEvent**

parts	<b>parameter</b>
	element <b>csta:AgentWorkingAfterCallEvent</b>
source	<code>&lt;message name="agentWorkingAfterCallEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AgentWorkingAfterCallEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **autoAnswerEvent**

parts	<b>parameter</b>
	element <b>csta:AutoAnswerEvent</b>
source	<code>&lt;message name="autoAnswerEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AutoAnswerEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **autoWorkModeEvent**

parts	<b>parameter</b>
	element <b>csta:AutoWorkModeEvent</b>
source	<code>&lt;message name="autoWorkModeEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:AutoWorkModeEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callBackEvent**

parts	<b>parameter</b>
	element <b>csta:CallBackEvent</b>
source	<code>&lt;message name="callBackEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBackEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callBackMessageEvent**

parts	<b>parameter</b>
	element <b>csta:CallBackMessageEvent</b>
source	<code>&lt;message name="callBackMessageEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:CallBackMessageEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message **callerIDStatusEvent**

parts	<b>parameter</b>
	element <b>csta:CallerIDStatusEvent</b>
source	<code>&lt;message name="callerIDStatusEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:CallerIDStatusEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message doNotDisturbEvent

parts	<b>parameter</b>
	element <b>csta:DoNotDisturbEvent</b>
source	<code>&lt;message name="doNotDisturbEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:DoNotDisturbEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message forwardingEvent

parts	<b>parameter</b>
	element <b>csta:ForwardingEvent</b>
source	<code>&lt;message name="forwardingEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:ForwardingEvent"/&gt;</code> <code>&lt;/message&gt;</code>

### message routeingEvent

parts	<b>parameter</b>
	element <b>csta:RouteingEvent</b>
source	<code>&lt;message name="routeingEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:RouteingEvent"/&gt;</code> <code>&lt;/message&gt;</code>

## 16.2 Operations

operations

### **CSTA-Call-Back-Message-Non-Call-Related**

input [tns:callBackMessageNonCallRelated](#)  
output [tns:callBackMessageNonCallRelatedResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Call-Back-Non-Call-Related**

input [tns:callBackNonCallRelated](#)  
output [tns:callBackNonCallRelatedResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Cancel-Call-Back**

input [tns:cancelCallBack](#)  
output [tns:cancelCallBackResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Cancel-Call-Back-Message**

input [tns:cancelCallBackMessage](#)  
output [tns:cancelCallBackMessageResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Get-Agent-State**

input [tns:getAgentState](#)  
output [tns:getAgentStateResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Get-Auto-Answer**

input [tns:getAutoAnswer](#)  
output [tns:getAutoAnswerResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Get-Auto-Work-Mode**

input [tns:getAutoWorkMode](#)  
output [tns:getAutoWorkModeResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Get-Caller-ID-Status**

input [tns:getCallerIDStatus](#)  
output [tns:getCallerIDStatusResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Get-Do-Not-Disturb**

input [tns:getDoNotDisturb](#)  
output [tns:getDoNotDisturbResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Get-Forwarding**

input [tns:getForwarding](#)  
output [tns:getForwardingResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Get-Last-Number-Dialed**

input [tns:getLastNumberDialed](#)  
output [tns:getLastNumberDialedResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Get-Routeing-Mode**

input [tns:getRouteingMode](#)  
output [tns:getRouteingModeResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Agent-State**

input [tns:setAgentState](#)  
output [tns:setAgentStateResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Auto-Answer**

input [tns:setAutoAnswer](#)  
output [tns:setAutoAnswerResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Auto-Work-Mode**

input [tns:setAutoWorkMode](#)  
output [tns:setAutoWorkModeResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Caller-ID-Status**

input [tns:setCallerIDStatus](#)  
output [tns:setCallerIDStatusResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Do-Not-Disturb**

input [tns:setDoNotDisturb](#)  
output [tns:setDoNotDisturbResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Set-Forwarding**

input [tns:setForwarding](#)  
output [tns:setForwardingResponse](#)  
fault [tns:negResponse](#)



**CSTA-Set-Routeing-Mode**

input [tns:setRouteingMode](#)  
output [tns:setRouteingModeResponse](#)  
fault [tns:negResponse](#)

**CSTA-Agent-Busy-Event**

output [tns:agentBusyEvent](#)

**CSTA-Agent-Logged-Off-Event**

output [tns:agentLoggedOffEvent](#)

**CSTA-Agent-Logged-On-Event**

output [tns:agentLoggedOnEvent](#)

**CSTA-Agent-Not-Ready-Event**

output [tns:agentNotReadyEvent](#)

**CSTA-Agent-Ready-Event**

output [tns:agentReadyEvent](#)

**CSTA-Agent-Working-After-Call-Event**

output [tns:agentWorkingAfterCallEvent](#)

**CSTA-Auto-Answer-Event**

output [tns:autoAnswerEvent](#)

**CSTA-Auto-Work-Mode-Event**

output [tns:autoWorkModeEvent](#)

**CSTA-Call-Back-Event**

output [tns:callBackEvent](#)

**CSTA-Call-Back-Message-Event**

output [tns:callBackMessageEvent](#)

**CSTA-Caller-ID-Status-Event**

output [tns:callerIDStatusEvent](#)

**CSTA-Do-Not-Disturb-Event**

output [tns:doNotDisturbEvent](#)

**CSTA-Forwarding-Event**

output [tns:forwardingEvent](#)

### CSTA-Routeing-Event

```
source      output tns:routeingEvent
<operation name="CSTA-Call-Back-Message-Non-Call-Related">
  <input message="tns:callBackMessageNonCallRelated"/>
  <output message="tns:callBackMessageNonCallRelatedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Call-Back-Non-Call-Related">
  <input message="tns:callBackNonCallRelated"/>
  <output message="tns:callBackNonCallRelatedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Cancel-Call-Back">
  <input message="tns:cancelCallBack"/>
  <output message="tns:cancelCallBackResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Cancel-Call-Back-Message">
  <input message="tns:cancelCallBackMessage"/>
  <output message="tns:cancelCallBackMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Agent-State">
  <input message="tns:getAgentState"/>
  <output message="tns:getAgentStateResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Auto-Answer">
  <input message="tns:getAutoAnswer"/>
  <output message="tns:getAutoAnswerResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Auto-Work-Mode">
  <input message="tns:getAutoWorkMode"/>
  <output message="tns:getAutoWorkModeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Caller-ID-Status">
  <input message="tns:getCallerIDStatus"/>
  <output message="tns:getCallerIDStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Do-Not-Disturb">
  <input message="tns:getDoNotDisturb"/>
```

```
<output message="tns:getDoNotDisturbResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Forwarding">
  <input message="tns:getForwarding"/>
  <output message="tns:getForwardingResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Last-Number-Dialed">
  <input message="tns:getLastNumberDialed"/>
  <output message="tns:getLastNumberDialedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Routeing-Mode">
  <input message="tns:getRouteingMode"/>
  <output message="tns:getRouteingModeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Agent-State">
  <input message="tns:setAgentState"/>
  <output message="tns:setAgentStateResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Auto-Answer">
  <input message="tns:setAutoAnswer"/>
  <output message="tns:setAutoAnswerResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Auto-Work-Mode">
  <input message="tns:setAutoWorkMode"/>
  <output message="tns:setAutoWorkModeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Caller-ID-Status">
  <input message="tns:setCallerIDStatus"/>
  <output message="tns:setCallerIDStatusResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Do-Not-Disturb">
  <input message="tns:setDoNotDisturb"/>
  <output message="tns:setDoNotDisturbResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Forwarding">
```

```
<input message="tns:setForwarding"/>
<output message="tns:setForwardingResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Routeing-Mode">
  <input message="tns:setRouteingMode"/>
  <output message="tns:setRouteingModeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Agent-Busy-Event">
  <output message="tns:agentBusyEvent"/>
</operation>
<operation name="CSTA-Agent-Logged-Off-Event">
  <output message="tns:agentLoggedOffEvent"/>
</operation>
<operation name="CSTA-Agent-Logged-On-Event">
  <output message="tns:agentLoggedOnEvent"/>
</operation>
<operation name="CSTA-Agent-Not-Ready-Event">
  <output message="tns:agentNotReadyEvent"/>
</operation>
<operation name="CSTA-Agent-Ready-Event">
  <output message="tns:agentReadyEvent"/>
</operation>
<operation name="CSTA-Agent-Working-After-Call-Event">
  <output message="tns:agentWorkingAfterCallEvent"/>
</operation>
<operation name="CSTA-Auto-Answer-Event">
  <output message="tns:autoAnswerEvent"/>
</operation>
<operation name="CSTA-Auto-Work-Mode-Event">
  <output message="tns:autoWorkModeEvent"/>
</operation>
<operation name="CSTA-Call-Back-Event">
  <output message="tns:callBackEvent"/>
</operation>
<operation name="CSTA-Call-Back-Message-Event">
  <output message="tns:callBackMessageEvent"/>
</operation>
<operation name="CSTA-Caller-ID-Status-Event">
  <output message="tns:callerIDStatusEvent"/>
</operation>
<operation name="CSTA-Do-Not-Disturb-Event">
  <output message="tns:doNotDisturbEvent"/>
</operation>
```

```
</operation>
<operation name="CSTA-Forwarding-Event">
  <output message="tns:forwardingEvent"/>
</operation>
<operation name="CSTA-Routeing-Event">
  <output message="tns:routeingEvent"/>
</operation>
```

## 17 Device Maintenance Events - Messages and Operations

This clause specifies the WSDL messages and operations for the device maintenance events.

### 17.1 Messages

#### message **backInServiceEvent**

```
parts    parameter
         element    csta:BackInServiceEvent
source   <message name="backInServiceEvent">
         <part name="parameter" element="csta:BackInServiceEvent"/>
         </message>
```

#### message **deviceCapsChangedEvent**

```
parts    parameter
         element    csta:DeviceCapsChangedEvent
source   <message name="deviceCapsChangedEvent">
         <part name="parameter" element="csta:DeviceCapsChangedEvent"/>
         </message>
```

#### message **outOfServiceEvent**

```
parts    parameter
         element    csta:OutOfServiceEvent
source   <message name="outOfServiceEvent">
         <part name="parameter" element="csta:OutOfServiceEvent"/>
         </message>
```

#### message **partiallyInServiceEvent**

```
parts    parameter
         element    csta:PartiallyInServiceEvent
source   <message name="partiallyInServiceEvent">
         <part name="parameter" element="csta:PartiallyInServiceEvent"/>
         </message>
```



## 17.2 Operations

operations	<p><b>CSTA-Back-In-Service-Event</b></p> <p>output <a href="#">tns:backInServiceEvent</a></p>
	<p><b>CSTA-Device-Capabilities-Changed-Event</b></p> <p>output <a href="#">tns:deviceCapsChangedEvent</a></p>
	<p><b>CSTA-Out-Of-Service-Event</b></p> <p>output <a href="#">tns:outOfServiceEvent</a></p>
	<p><b>CSTA-Partially-In-Service-Event</b></p> <p>output <a href="#">tns:partiallyInServiceEvent</a></p>
source	<pre> &lt;operation name="CSTA-Back-In-Service-Event"&gt;   &lt;output message="tns:backInServiceEvent"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Device-Capabilities-Changed-Event"&gt;   &lt;output message="tns:deviceCapsChangedEvent"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Out-Of-Service-Event"&gt;   &lt;output message="tns:outOfServiceEvent"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Partially-In-Service-Event"&gt;   &lt;output message="tns:partiallyInServiceEvent"/&gt; &lt;/operation&gt; </pre>

## 18 I/O Services Features - Messages and Operations

This clause specifies the WSDL messages and operations for the I/O Services features.

### 18.1 Messages

#### message iORegister

```
parts    parameter
         element    csta:IORegister
source   <message name="iORegister">
         <part name="parameter" element="csta:IORegister"/>
         </message>
```

#### message iORegisterResponse

```
parts    parameter
         element    csta:IORegisterResponse
source   <message name="iORegisterResponse">
         <part name="parameter" element="csta:IORegisterResponse"/>
         </message>
```

#### message iORegisterAbort

```
parts    parameter
         element    csta:IORegisterAbort
source   <message name="iORegisterAbort">
         <part name="parameter" element="csta:IORegisterAbort"/>
         </message>
```

#### message iORegisterCancel

```
parts    parameter
         element    csta:IORegisterCancel
source   <message name="iORegisterCancel">
         <part name="parameter" element="csta:IORegisterCancel"/>
         </message>
```



### message **IORegisterCancelResponse**

```
parts    parameter
         element  csta:IORegisterCancelResponse
source   <message name="IORegisterCancelResponse">
         <part name="parameter" element="csta:IORegisterCancelResponse"/>
         </message>
```

### message **dataPathResumed**

```
parts    parameter
         element  csta:DataPathResumed
source   <message name="dataPathResumed">
         <part name="parameter" element="csta:DataPathResumed"/>
         </message>
```

### message **dataPathResumedResponse**

```
parts    parameter
         element  csta:DataPathResumedResponse
source   <message name="dataPathResumedResponse">
         <part name="parameter" element="csta:DataPathResumedResponse"/>
         </message>
```

### message **dataPathSuspended**

```
parts    parameter
         element  csta:DataPathSuspended
source   <message name="dataPathSuspended">
         <part name="parameter" element="csta:DataPathSuspended"/>
         </message>
```

### message **dataPathSuspendedResponse**

```
parts    parameter
         element  csta:DataPathSuspendedResponse
source   <message name="dataPathSuspendedResponse">
         <part name="parameter" element="csta:DataPathSuspendedResponse"/>
         </message>
```

### message **fastData**

```
parts      parameter
           element   csta:FastData
source    <message name="fastData">
           <part name="parameter" element="csta:FastData"/>
           </message>
```

### message **fastDataResponse**

```
parts      parameter
           element   csta:FastDataResponse
source    <message name="fastDataResponse">
           <part name="parameter" element="csta:FastDataResponse"/>
           </message>
```

### message **resumeDataPath**

```
parts      parameter
           element   csta:ResumeDataPath
source    <message name="resumeDataPath">
           <part name="parameter" element="csta:ResumeDataPath"/>
           </message>
```

### message **resumeDataPathResponse**

```
parts      parameter
           element   csta:ResumeDataPathResponse
source    <message name="resumeDataPathResponse">
           <part name="parameter" element="csta:ResumeDataPathResponse"/>
           </message>
```

### message **sendBroadcastData**

```
parts    parameter
         element  csta:SendBroadcastData
source   <message name="sendBroadcastData">
         <part name="parameter" element="csta:SendBroadcastData"/>
         </message>
```

### message **sendBroadcastDataResponse**

```
parts    parameter
         element  csta:SendBroadcastDataResponse
source   <message name="sendBroadcastDataResponse">
         <part name="parameter" element="csta:SendBroadcastDataResponse"/>
         </message>
```

### message **sendData**

```
parts    parameter
         element  csta:SendData
source   <message name="sendData">
         <part name="parameter" element="csta:SendData"/>
         </message>
```

### message **sendDataResponse**

```
parts    parameter
         element  csta:SendDataResponse
source   <message name="sendDataResponse">
         <part name="parameter" element="csta:SendDataResponse"/>
         </message>
```

### message **sendMulticastData**

```
parts    parameter
         element  csta:SendMulticastData
source   <message name="sendMulticastData">
         <part name="parameter" element="csta:SendMulticastData"/>
         </message>
```

### message **sendMulticastDataResponse**

parts	<b>parameter</b>
	element <b>csta:SendMulticastDataResponse</b>
source	<code>&lt;message name="sendMulticastDataResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SendMulticastDataResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **startDataPath**

parts	<b>parameter</b>
	element <b>csta:StartDataPath</b>
source	<code>&lt;message name="startDataPath"&gt;</code> <code>&lt;part name="parameter" element="csta:StartDataPath"/&gt;</code> <code>&lt;/message&gt;</code>

### message **startDataPathResponse**

parts	<b>parameter</b>
	element <b>csta:StartDataPathResponse</b>
source	<code>&lt;message name="startDataPathResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:StartDataPathResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **stopDataPath**

parts	<b>parameter</b>
	element <b>csta:StopDataPath</b>
source	<code>&lt;message name="stopDataPath"&gt;</code> <code>&lt;part name="parameter" element="csta:StopDataPath"/&gt;</code> <code>&lt;/message&gt;</code>

### message **stopDataPathResponse**

```
parts    parameter
        element csta:StopDataPathResponse
source  <message name="stopDataPathResponse">
        <part name="parameter" element="csta:StopDataPathResponse"/>
        </message>
```

### message **suspendDataPath**

```
parts    parameter
        element csta:SuspendDataPath
source  <message name="suspendDataPath">
        <part name="parameter" element="csta:SuspendDataPath"/>
        </message>
```

### message **suspendDataPathResponse**

```
parts    parameter
        element csta:SuspendDataPathResponse
source  <message name="suspendDataPathResponse">
        <part name="parameter" element="csta:SuspendDataPathResponse"/>
        </message>
```

## 18.2 Operations

operations

### CSTA-IO-Register

input [tns:iORegister](#)  
output [tns:iORegisterResponse](#)  
fault [tns:negResponse](#)

### CSTA-IO-Register-Abort

output [tns:iORegisterAbort](#)

### CSTA-IO-Register-Cancel

input [tns:iORegisterCancel](#)  
output [tns:iORegisterCancelResponse](#)  
fault [tns:negResponse](#)

### CSTA-Data-Path-Resumed

output [tns:dataPathResumed](#)  
input [tns:dataPathResumedResponse](#)  
fault [tns:negResponse](#)

### CSTA-Data-Path-Suspended

output [tns:dataPathSuspended](#)  
input [tns:dataPathSuspendedResponse](#)  
fault [tns:negResponse](#)

### CSTA-Fast-Data-as-input

input [tns:fastData](#)  
output [tns:fastDataResponse](#)  
fault [tns:negResponse](#)

### CSTA-Fast-Data-as-output

output [tns:fastData](#)  
input [tns:fastDataResponse](#)  
fault [tns:negResponse](#)

### CSTA-Resume-Data-Path

input [tns:resumeDataPath](#)  
output [tns:resumeDataPathResponse](#)  
fault [tns:negResponse](#)

### CSTA-Send-Broadcast-Data

input [tns:sendBroadcastData](#)  
output [tns:sendBroadcastDataResponse](#)  
fault [tns:negResponse](#)

**CSTA-Send-Data-as-input**

input [tns:sendData](#)  
 output [tns:sendDataResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Send-Data-as-output**

output [tns:sendData](#)  
 input [tns:sendDataResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Send-Multicast-Data**

input [tns:sendMulticastData](#)  
 output [tns:sendMulticastDataResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Start-Data-Path-as-input**

input [tns:startDataPath](#)  
 output [tns:startDataPathResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Start-Data-Path-as-output**

output [tns:startDataPath](#)  
 input [tns:startDataPathResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Stop-Data-Path-as-input**

input [tns:stopDataPath](#)  
 output [tns:stopDataPathResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Stop-Data-Path-as-output**

output [tns:stopDataPath](#)  
 input [tns:stopDataPathResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Suspend-Data-Path-as-input**

input [tns:suspendDataPath](#)  
 output [tns:suspendDataPathResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Suspend-Data-Path-as-output**

output [tns:suspendDataPath](#)  
 input [tns:suspendDataPathResponse](#)  
 fault [tns:negResponse](#)

source `<operation name="CSTA-IO-Register">  
 <input message="tns:iORegister"/>  
 <output message="tns:iORegisterResponse"/>`

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-IO-Register-Abort">
  <output message="tns:iORegisterAbort"/>
</operation>
<operation name="CSTA-IO-Register-Cancel">
  <input message="tns:iORegisterCancel"/>
  <output message="tns:iORegisterCancelResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Data-Path-Resumed">
  <output message="tns:dataPathResumed"/>
  <input message="tns:dataPathResumedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Data-Path-Suspended">
  <output message="tns:dataPathSuspended"/>
  <input message="tns:dataPathSuspendedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Fast-Data-as-input">
  <input message="tns:fastData"/>
  <output message="tns:fastDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Fast-Data-as-output">
  <output message="tns:fastData"/>
  <input message="tns:fastDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Resume-Data-Path">
  <input message="tns:resumeDataPath"/>
  <output message="tns:resumeDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-Broadcast-Data">
  <input message="tns:sendBroadcastData"/>
  <output message="tns:sendBroadcastDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-Data-as-input">
  <input message="tns:sendData"/>
  <output message="tns:sendDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
```



```
</operation>
<operation name="CSTA-Send-Data-as-output">
  <output message="tns:sendData"/>
  <input message="tns:sendDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-Multicast-Data">
  <input message="tns:sendMulticastData"/>
  <output message="tns:sendMulticastDataResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Start-Data-Path-as-input">
  <input message="tns:startDataPath"/>
  <output message="tns:startDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Start-Data-Path-as-output">
  <output message="tns:startDataPath"/>
  <input message="tns:startDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Stop-Data-Path-as-input">
  <input message="tns:stopDataPath"/>
  <output message="tns:stopDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Stop-Data-Path-as-output">
  <output message="tns:stopDataPath"/>
  <input message="tns:stopDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Suspend-Data-Path-as-input">
  <input message="tns:suspendDataPath"/>
  <output message="tns:suspendDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Suspend-Data-Path-as-output">
  <output message="tns:suspendDataPath"/>
  <input message="tns:suspendDataPathResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

## 19 Data Collection Features - Messages and Operations

This clause specifies the WSDL messages and operations for the data collection features.

### 19.1 Messages

#### message **dataCollected**

```
parts    parameter
         element    csta:DataCollected
source  <message name="dataCollected">
         <part name="parameter" element="csta:DataCollected"/>
         </message>
```

#### message **dataCollectedResponse**

```
parts    parameter
         element    csta:DataCollectedResponse
source  <message name="dataCollectedResponse">
         <part name="parameter" element="csta:DataCollectedResponse"/>
         </message>
```

#### message **dataCollectionResumed**

```
parts    parameter
         element    csta:DataCollectionResumed
source  <message name="dataCollectionResumed">
         <part name="parameter" element="csta:DataCollectionResumed"/>
         </message>
```

#### message **dataCollectionResumedResponse**

```
parts    parameter
         element    csta:DataCollectionResumedResponse
source  <message name="dataCollectionResumedResponse">
         <part name="parameter" element="csta:DataCollectionResumedResponse"/>
         </message>
```

### message **dataCollectionSuspended**

```
parts    parameter
         element  csta:DataCollectionSuspended
source   <message name="dataCollectionSuspended">
         <part name="parameter" element="csta:DataCollectionSuspended"/>
         </message>
```

### message **dataCollectionSuspendedResponse**

```
parts    parameter
         element  csta:DataCollectionSuspendedResponse
source   <message name="dataCollectionSuspendedResponse">
         <part name="parameter" element="csta:DataCollectionSuspendedResponse"/>
         </message>
```

### message **resumeDataCollection**

```
parts    parameter
         element  csta:ResumeDataCollection
source   <message name="resumeDataCollection">
         <part name="parameter" element="csta:ResumeDataCollection"/>
         </message>
```

### message **resumeDataCollectionResponse**

```
parts    parameter
         element  csta:ResumeDataCollectionResponse
source   <message name="resumeDataCollectionResponse">
         <part name="parameter" element="csta:ResumeDataCollectionResponse"/>
         </message>
```

### message **startDataCollection**

```
parts    parameter
         element  csta:StartDataCollection
source   <message name="startDataCollection">
         <part name="parameter" element="csta:StartDataCollection"/>
         </message>
```

### message **startDataCollectionResponse**

```
parts    parameter
         element  csta:StartDataCollectionResponse
source  <message name="startDataCollectionResponse">
         <part name="parameter" element="csta:StartDataCollectionResponse"/>
         </message>
```

### message **stopDataCollection**

```
parts    parameter
         element  csta:StopDataCollection
source  <message name="stopDataCollection">
         <part name="parameter" element="csta:StopDataCollection"/>
         </message>
```

### message **stopDataCollectionResponse**

```
parts    parameter
         element  csta:StopDataCollectionResponse
source  <message name="stopDataCollectionResponse">
         <part name="parameter" element="csta:StopDataCollectionResponse"/>
         </message>
```

### message **suspendDataCollection**

```
parts    parameter
         element  csta:SuspendDataCollection
source  <message name="suspendDataCollection">
         <part name="parameter" element="csta:SuspendDataCollection"/>
         </message>
```

## message **suspendDataCollectionResponse**

parts	<b>parameter</b>
	element <b>csta:SuspendDataCollectionResponse</b>
source	<code>&lt;message name="suspendDataCollectionResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SuspendDataCollectionResponse"/&gt;</code> <code>&lt;/message&gt;</code>

## 19.2 Operations

operations

### CSTA-Data-Collected

output [tns:dataCollected](#)  
input [tns:dataCollectedResponse](#)  
fault [tns:negResponse](#)

### CSTA-Data-Collection-Resumed

output [tns:dataCollectionResumed](#)  
input [tns:dataCollectionResumedResponse](#)  
fault [tns:negResponse](#)

### CSTA-Data-Collection-Suspended

output [tns:dataCollectionSuspended](#)  
input [tns:dataCollectionSuspendedResponse](#)  
fault [tns:negResponse](#)

### CSTA-Resume-Data-Collection

input [tns:resumeDataCollection](#)  
output [tns:resumeDataCollectionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Start-Data-Collection

input [tns:startDataCollection](#)  
output [tns:startDataCollectionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Stop-Data-Collection-as-input

input [tns:stopDataCollection](#)  
output [tns:stopDataCollectionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Stop-Data-Collection-as-output

output [tns:stopDataCollection](#)  
input [tns:stopDataCollectionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Suspend-Data-Collection-as-input

input [tns:suspendDataCollection](#)  
output [tns:suspendDataCollectionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Suspend-Data-Collection-as-output

output [tns:suspendDataCollection](#)  
input [tns:suspendDataCollectionResponse](#)  
fault [tns:negResponse](#)

source

<operation name="CSTA-Data-Collected">

```
<output message="tns:dataCollected"/>
<input message="tns:dataCollectedResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Data-Collection-Resumed">
  <output message="tns:dataCollectionResumed"/>
  <input message="tns:dataCollectionResumedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Data-Collection-Suspended">
  <output message="tns:dataCollectionSuspended"/>
  <input message="tns:dataCollectionSuspendedResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Resume-Data-Collection">
  <input message="tns:resumeDataCollection"/>
  <output message="tns:resumeDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Start-Data-Collection">
  <input message="tns:startDataCollection"/>
  <output message="tns:startDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Stop-Data-Collection-as-input">
  <input message="tns:stopDataCollection"/>
  <output message="tns:stopDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Stop-Data-Collection-as-output">
  <output message="tns:stopDataCollection"/>
  <input message="tns:stopDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Suspend-Data-Collection-as-input">
  <input message="tns:suspendDataCollection"/>
  <output message="tns:suspendDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Suspend-Data-Collection-as-output">
  <output message="tns:suspendDataCollection"/>
  <input message="tns:suspendDataCollectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

## 20 Voice Features - Messages and Operations

This clause specifies the WSDL messages and operations for the Voice Services.

### 20.1 Messages

#### message **activate**

```
parts    parameter
         element    csta:Activate
source  <message name="activate">
         <part name="parameter" element="csta:Activate"/>
         </message>
```

#### message **activateResponse**

```
parts    parameter
         element    csta:ActivateResponse
source  <message name="activateResponse">
         <part name="parameter" element="csta:ActivateResponse"/>
         </message>
```

#### message **clear**

```
parts    parameter
         element    csta:Clear
source  <message name="clear">
         <part name="parameter" element="csta:Clear"/>
         </message>
```

#### message **clearResponse**

```
parts    parameter
         element    csta:ClearResponse
source  <message name="clearResponse">
         <part name="parameter" element="csta:ClearResponse"/>
         </message>
```



### message concatenateMessage

```
parts    parameter
         element  csta:ConcatenateMessage
source   <message name="concatenateMessage">
         <part name="parameter" element="csta:ConcatenateMessage"/>
         </message>
```

### message concatenateMessageResponse

```
parts    parameter
         element  csta:ConcatenateMessageResponse
source   <message name="concatenateMessageResponse">
         <part name="parameter" element="csta:ConcatenateMessageResponse"/>
         </message>
```

### message deactivate

```
parts    parameter
         element  csta:Deactivate
source   <message name="deactivate">
         <part name="parameter" element="csta:Deactivate"/>
         </message>
```

### message deactivateResponse

```
parts    parameter
         element  csta:DeactivateResponse
source   <message name="deactivateResponse">
         <part name="parameter" element="csta:DeactivateResponse"/>
         </message>
```

### message deleteMessage

```
parts    parameter
         element  csta>DeleteMessage
source   <message name="deleteMessage">
         <part name="parameter" element="csta>DeleteMessage"/>
         </message>
```

### message deleteMessageResponse

parts	<b>parameter</b>
	element <b>csta&gt;DeleteMessageResponse</b>
source	<code>&lt;message name="deleteMessageResponse"&gt;</code> <code>&lt;part name="parameter" element="csta&gt;DeleteMessageResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message playMessage

parts	<b>parameter</b>
	element <b>csta:PlayMessage</b>
source	<code>&lt;message name="playMessage"&gt;</code> <code>&lt;part name="parameter" element="csta:PlayMessage"/&gt;</code> <code>&lt;/message&gt;</code>

### message playMessageResponse

parts	<b>parameter</b>
	element <b>csta:PlayMessageResponse</b>
source	<code>&lt;message name="playMessageResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:PlayMessageResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message queryVoiceAttribute

parts	<b>parameter</b>
	element <b>csta:QueryVoiceAttribute</b>
source	<code>&lt;message name="queryVoiceAttribute"&gt;</code> <code>&lt;part name="parameter" element="csta:QueryVoiceAttribute"/&gt;</code> <code>&lt;/message&gt;</code>

### message queryVoiceAttributeResponse

```
parts    parameter
         element  csta:QueryVoiceAttributeResponse
source   <message name="queryVoiceAttributeResponse">
         <part name="parameter" element="csta:QueryVoiceAttributeResponse"/>
         </message>
```

### message queueService

```
parts    parameter
         element  csta:QueueService
source   <message name="queueService">
         <part name="parameter" element="csta:QueueService"/>
         </message>
```

### message queueServiceResponse

```
parts    parameter
         element  csta:QueueServiceResponse
source   <message name="queueServiceResponse">
         <part name="parameter" element="csta:QueueServiceResponse"/>
         </message>
```

### message recordMessage

```
parts    parameter
         element  csta:RecordMessage
source   <message name="recordMessage">
         <part name="parameter" element="csta:RecordMessage"/>
         </message>
```

### message recordMessageResponse

```
parts    parameter
         element  csta:RecordMessageResponse
source   <message name="recordMessageResponse">
         <part name="parameter" element="csta:RecordMessageResponse"/>
         </message>
```

### message reposition

parts	<b>parameter</b>
	element <b>csta:Reposition</b>
source	<code>&lt;message name="reposition"&gt;</code> <code>&lt;part name="parameter" element="csta:Reposition"/&gt;</code> <code>&lt;/message&gt;</code>

### message repositionResponse

parts	<b>parameter</b>
	element <b>csta:RepositionResponse</b>
source	<code>&lt;message name="repositionResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:RepositionResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message resume

parts	<b>parameter</b>
	element <b>csta:Resume</b>
source	<code>&lt;message name="resume"&gt;</code> <code>&lt;part name="parameter" element="csta:Resume"/&gt;</code> <code>&lt;/message&gt;</code>

### message resumeResponse

parts	<b>parameter</b>
	element <b>csta:ResumeResponse</b>
source	<code>&lt;message name="resumeResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:ResumeResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message review

```
parts    parameter
         element  csta:Review
source   <message name="review">
         <part name="parameter" element="csta:Review"/>
         </message>
```

### message reviewResponse

```
parts    parameter
         element  csta:ReviewResponse
source   <message name="reviewResponse">
         <part name="parameter" element="csta:ReviewResponse"/>
         </message>
```

### message setVoiceAttribute

```
parts    parameter
         element  csta:SetVoiceAttribute
source   <message name="setVoiceAttribute">
         <part name="parameter" element="csta:SetVoiceAttribute"/>
         </message>
```

### message setVoiceAttributeResponse

```
parts    parameter
         element  csta:SetVoiceAttributeResponse
source   <message name="setVoiceAttributeResponse">
         <part name="parameter" element="csta:SetVoiceAttributeResponse"/>
         </message>
```

### message start

```
parts    parameter
         element  csta:Start
source   <message name="start">
         <part name="parameter" element="csta:Start"/>
         </message>
```

### message **startResponse**

```
parts    parameter
         element csta:StartResponse
source  <message name="startResponse">
         <part name="parameter" element="csta:StartResponse"/>
       </message>
```

### message **stop**

```
parts    parameter
         element csta:Stop
source  <message name="stop">
         <part name="parameter" element="csta:Stop"/>
       </message>
```

### message **stopResponse**

```
parts    parameter
         element csta:StopResponse
source  <message name="stopResponse">
         <part name="parameter" element="csta:StopResponse"/>
       </message>
```

### message **suspend**

```
parts    parameter
         element csta:Suspend
source  <message name="suspend">
         <part name="parameter" element="csta:Suspend"/>
       </message>
```

### message **suspendResponse**

```
parts    parameter
         element  csta:SuspendResponse
source  <message name="suspendResponse">
         <part name="parameter" element="csta:SuspendResponse"/>
         </message>
```

### message **synthesizeMessage**

```
parts    parameter
         element  csta:SynthesizeMessage
source  <message name="synthesizeMessage">
         <part name="parameter" element="csta:SynthesizeMessage"/>
         </message>
```

### message **synthesizeMessageResponse**

```
parts    parameter
         element  csta:SynthesizeMessageResponse
source  <message name="synthesizeMessageResponse">
         <part name="parameter" element="csta:SynthesizeMessageResponse"/>
         </message>
```

### message **bookmarkReachedEvent**

```
parts    parameter
         element  csta:BookmarkReachedEvent
source  <message name="bookmarkReachedEvent">
         <part name="parameter" element="csta:BookmarkReachedEvent"/>
         </message>
```

### message **completedEvent**

```
parts    parameter
         element  csta:CompletedEvent
source  <message name="completedEvent">
         <part name="parameter" element="csta:CompletedEvent"/>
         </message>
```

### message dtmfDetectedEvent

```
parts      parameter
           element  csta:DtmfDetectedEvent
source     <message name="dtmfDetectedEvent">
           <part name="parameter" element="csta:DtmfDetectedEvent"/>
           </message>
```

### message emptiedEvent

```
parts      parameter
           element  csta:EmptiedEvent
source     <message name="emptiedEvent">
           <part name="parameter" element="csta:EmptiedEvent"/>
           </message>
```

### message interruptionDetectedEvent

```
parts      parameter
           element  csta:InterruptionDetectedEvent
source     <message name="interruptionDetectedEvent">
           <part name="parameter" element="csta:InterruptionDetectedEvent"/>
           </message>
```

### message notRecognizedEvent

```
parts      parameter
           element  csta:NotRecognizedEvent
source     <message name="notRecognizedEvent">
           <part name="parameter" element="csta:NotRecognizedEvent"/>
           </message>
```



### message **playEvent**

```
parts    parameter
         element csta:PlayEvent
source   <message name="playEvent">
         <part name="parameter" element="csta:PlayEvent"/>
         </message>
```

### message **recognizedEvent**

```
parts    parameter
         element csta:RecognizedEvent
source   <message name="recognizedEvent">
         <part name="parameter" element="csta:RecognizedEvent"/>
         </message>
```

### message **recordEvent**

```
parts    parameter
         element csta:RecordEvent
source   <message name="recordEvent">
         <part name="parameter" element="csta:RecordEvent"/>
         </message>
```

### message **reviewEvent**

```
parts    parameter
         element csta:ReviewEvent
source   <message name="reviewEvent">
         <part name="parameter" element="csta:ReviewEvent"/>
         </message>
```

### message **silenceTimeoutExpiredEvent**

```
parts    parameter
         element csta:SilenceTimeoutExpiredEvent
source   <message name="silenceTimeoutExpiredEvent">
         <part name="parameter" element="csta:SilenceTimeoutExpiredEvent"/>
         </message>
```

### message **speechDetectedEvent**

```
parts      parameter
           element  csta:SpeechDetectedEvent
source     <message name="speechDetectedEvent">
           <part name="parameter" element="csta:SpeechDetectedEvent"/>
           </message>
```

### message **startedEvent**

```
parts      parameter
           element  csta:StartedEvent
source     <message name="startedEvent">
           <part name="parameter" element="csta:StartedEvent"/>
           </message>
```

### message **stopEvent**

```
parts      parameter
           element  csta:StopEvent
source     <message name="stopEvent">
           <part name="parameter" element="csta:StopEvent"/>
           </message>
```

### message **suspendPlayEvent**

```
parts      parameter
           element  csta:SuspendPlayEvent
source     <message name="suspendPlayEvent">
           <part name="parameter" element="csta:SuspendPlayEvent"/>
           </message>
```

### message **suspendRecordEvent**

```
parts    parameter
         element csta:SuspendRecordEvent
source  <message name="suspendRecordEvent">
        <part name="parameter" element="csta:SuspendRecordEvent"/>
        </message>
```

### message **voiceAttributeChangedEvent**

```
parts    parameter
         element csta:VoiceAttributeChangedEvent
source  <message name="voiceAttributeChangedEvent">
        <part name="parameter" element="csta:VoiceAttributeChangedEvent"/>
        </message>
```

### message **voiceErrorOccurredEvent**

```
parts    parameter
         element csta:VoiceErrorOccurredEvent
source  <message name="voiceErrorOccurredEvent">
        <part name="parameter" element="csta:VoiceErrorOccurredEvent"/>
        </message>
```

## 20.2 Operations

operations

### **CSTA-Activate**

input [tns:activate](#)  
output [tns:activateResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Clear**

input [tns:clear](#)  
output [tns:clearResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Concatenate-Message**

input [tns:concatenateMessage](#)  
output [tns:concatenateMessageResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Deactivate**

input [tns:deactivate](#)  
output [tns:deactivateResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Delete-Message**

input [tns:deleteMessage](#)  
output [tns:deleteMessageResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Play-Message**

input [tns:playMessage](#)  
output [tns:playMessageResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Query-Voice-Attribute**

input [tns:queryVoiceAttribute](#)  
output [tns:queryVoiceAttributeResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Queue-Service**

input [tns:queueService](#)  
output [tns:queueServiceResponse](#)  
fault [tns:negResponse](#)

### **CSTA-Record-Message**

input [tns:recordMessage](#)  
output [tns:recordMessageResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Reposition**

input [tms:reposition](#)  
output [tms:repositionResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Resume**

input [tms:resume](#)  
output [tms:resumeResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Review**

input [tms:review](#)  
output [tms:reviewResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Set-Voice-Attribute**

input [tms:setVoiceAttribute](#)  
output [tms:setVoiceAttributeResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Start**

input [tms:start](#)  
output [tms:startResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Stop**

input [tms:stop](#)  
output [tms:stopResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Suspend**

input [tms:suspend](#)  
output [tms:suspendResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Synthesize-Message**

input [tms:synthesizeMessage](#)  
output [tms:synthesizeMessageResponse](#)  
fault [tms:negResponse](#)

#### **CSTA-Bookmark-Reached-Event**

output [tms:bookmarkReachedEvent](#)

#### **CSTA-Completed-Event**

output [tms:completedEvent](#)

**CSTA-Dtmf-Detected-Event**

output [tns:dtmfDetectedEvent](#)

**CSTA-Emptied-Event**

output [tns:emptiedEvent](#)

**CSTA-Interruption-Detected-Event**

output [tns:interruptionDetectedEvent](#)

**CSTA-Not-Recognized-Event**

output [tns:notRecognizedEvent](#)

**CSTA-Play-Event**

output [tns:playEvent](#)

**CSTA-Recognized-Event**

output [tns:recognizedEvent](#)

**CSTA-Record-Event**

output [tns:recordEvent](#)

**CSTA-Review-Event**

output [tns:reviewEvent](#)

**CSTA-Silence-Timeout-Expired-Event**

output [tns:silenceTimeoutExpiredEvent](#)

**CSTA-Speech-Detected-Event**

output [tns:speechDetectedEvent](#)

**CSTA-Started-Event**

output [tns:startedEvent](#)

**CSTA-Stop-Event**

output [tns:stopEvent](#)

**CSTA-Suspend-Play-Event**

output [tns:suspendPlayEvent](#)

**CSTA-Suspend-Record-Event**

output [tns:suspendRecordEvent](#)

**CSTA-Voice-Attribute-Changed-Event**

output [tns:voiceAttributeChangedEvent](#)

### CSTA-Voice-Error-Occurred-Event

output [tns:voiceErrorOccurredEvent](#)

```

source <operation name="CSTA-Activate">
  <input message="tns:activate"/>
  <output message="tns:activateResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear">
  <input message="tns:clear"/>
  <output message="tns:clearResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Concatenate-Message">
  <input message="tns:concatenateMessage"/>
  <output message="tns:concatenateMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deactivate">
  <input message="tns:deactivate"/>
  <output message="tns:deactivateResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Delete-Message">
  <input message="tns:deleteMessage"/>
  <output message="tns:deleteMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Play-Message">
  <input message="tns:playMessage"/>
  <output message="tns:playMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Query-Voice-Attribute">
  <input message="tns:queryVoiceAttribute"/>
  <output message="tns:queryVoiceAttributeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Queue-Service">
  <input message="tns:queueService"/>
  <output message="tns:queueServiceResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Record-Message">

```

```
<input message="tns:recordMessage"/>
<output message="tns:recordMessageResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reposition">
  <input message="tns:reposition"/>
  <output message="tns:repositionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Resume">
  <input message="tns:resume"/>
  <output message="tns:resumeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Review">
  <input message="tns:review"/>
  <output message="tns:reviewResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Set-Voice-Attribute">
  <input message="tns:setVoiceAttribute"/>
  <output message="tns:setVoiceAttributeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Start">
  <input message="tns:start"/>
  <output message="tns:startResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Stop">
  <input message="tns:stop"/>
  <output message="tns:stopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Suspend">
  <input message="tns:suspend"/>
  <output message="tns:suspendResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Synthesize-Message">
  <input message="tns:synthesizeMessage"/>
  <output message="tns:synthesizeMessageResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```



```
<operation name="CSTA-Bookmark-Reached-Event">
  <output message="tns:bookmarkReachedEvent"/>
</operation>
<operation name="CSTA-Completed-Event">
  <output message="tns:completedEvent"/>
</operation>
<operation name="CSTA-Dtmf-Dectected-Event">
  <output message="tns:dtmfDetectedEvent"/>
</operation>
<operation name="CSTA-Emptied-Event">
  <output message="tns:emptiedEvent"/>
</operation>
<operation name="CSTA-Interruption-Detected-Event">
  <output message="tns:interruptionDetectedEvent"/>
</operation>
<operation name="CSTA-Not-Recognized-Event">
  <output message="tns:notRecognizedEvent"/>
</operation>
<operation name="CSTA-Play-Event">
  <output message="tns:playEvent"/>
</operation>
<operation name="CSTA-Recognized-Event">
  <output message="tns:recognizedEvent"/>
</operation>
<operation name="CSTA-Record-Event">
  <output message="tns:recordEvent"/>
</operation>
<operation name="CSTA-Review-Event">
  <output message="tns:reviewEvent"/>
</operation>
<operation name="CSTA-Silence-Timeout-Expired-Event">
  <output message="tns:silenceTimeoutExpiredEvent"/>
</operation>
<operation name="CSTA-Speech-Detected-Event">
  <output message="tns:speechDetectedEvent"/>
</operation>
<operation name="CSTA-Started-Event">
  <output message="tns:startedEvent"/>
</operation>
<operation name="CSTA-Stop-Event">
  <output message="tns:stopEvent"/>
</operation>
<operation name="CSTA-Suspend-Play-Event">
  <output message="tns:suspendPlayEvent"/>
```

```
</operation>
<operation name="CSTA-Suspend-Record-Event">
  <output message="tns:suspendRecordEvent"/>
</operation>
<operation name="CSTA-Voice-Attribute-Changed-Event">
  <output message="tns:voiceAttributeChangedEvent"/>
</operation>
<operation name="CSTA-Voice-Error-Occurred-Event">
  <output message="tns:voiceErrorOccurredEvent"/>
</operation>
```

## 21 Call Data Recording Features - Messages and Operations

This clause specifies the WSDL messages and operations for the Call Data Recording features.

### 21.1 Messages

#### message **cDRNotification**

```
parts    parameter
         element  csta:CDRNotification
source   <message name="cDRNotification">
         <part name="parameter" element="csta:CDRNotification"/>
         </message>
```

#### message **cDRNotificationResponse**

```
parts    parameter
         element  csta:CDRNotificationResponse
source   <message name="cDRNotificationResponse">
         <part name="parameter" element="csta:CDRNotificationResponse"/>
         </message>
```

#### message **cDRReport**

```
parts    parameter
         element  csta:CDRReport
source   <message name="cDRReport">
         <part name="parameter" element="csta:CDRReport"/>
         </message>
```

#### message **cDRReportResponse**

```
parts    parameter
         element  csta:CDRReportResponse
source   <message name="cDRReportResponse">
         <part name="parameter" element="csta:CDRReportResponse"/>
         </message>
```

### message **sendStoredCDR**

parts	<b>parameter</b>
	element <b>csta:SendStoredCDR</b>
source	<code>&lt;message name="sendStoredCDR"&gt;</code> <code>&lt;part name="parameter" element="csta:SendStoredCDR"/&gt;</code> <code>&lt;/message&gt;</code>

### message **sendStoredCDRResponse**

parts	<b>parameter</b>
	element <b>csta:SendStoredCDRResponse</b>
source	<code>&lt;message name="sendStoredCDRResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:SendStoredCDRResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **startCDRTransmission**

parts	<b>parameter</b>
	element <b>csta:StartCDRTransmission</b>
source	<code>&lt;message name="startCDRTransmission"&gt;</code> <code>&lt;part name="parameter" element="csta:StartCDRTransmission"/&gt;</code> <code>&lt;/message&gt;</code>

### message **startCDRTransmissionResponse**

parts	<b>parameter</b>
	element <b>csta:StartCDRTransmissionResponse</b>
source	<code>&lt;message name="startCDRTransmissionResponse"&gt;</code> <code>&lt;part name="parameter" element="csta:StartCDRTransmissionResponse"/&gt;</code> <code>&lt;/message&gt;</code>

### message **stopCDRTransmission**

parts	<b>parameter</b>
	element <b>csta:StopCDRTransmission</b>
source	<code>&lt;message name="stopCDRTransmission"&gt;</code> <code>&lt;part name="parameter" element="csta:StopCDRTransmission"/&gt;</code> <code>&lt;/message&gt;</code>

### message **stopCDRTransmissionResponse**

```
parts    parameter
        element csta:StopCDRTransmissionResponse
source  <message name="stopCDRTransmissionResponse">
        <part name="parameter" element="csta:StopCDRTransmissionResponse"/>
        </message>
```

## 21.2 Operations

operations

### CSTA-CDR-Notification

output [tns:cDRNotification](#)  
input [tns:cDRNotificationResponse](#)  
fault [tns:negResponse](#)

### CSTA-CDR-Report

output [tns:cDRReport](#)  
input [tns:cDRReportResponse](#)  
fault [tns:negResponse](#)

### CSTA-Send-Stored-CDR

input [tns:sendStoredCDR](#)  
output [tns:sendStoredCDRResponse](#)  
fault [tns:negResponse](#)

### CSTA-Start-CDR-Transmission

input [tns:startCDRTransmission](#)  
output [tns:startCDRTransmissionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Stop-CDR-Transmission-as-input

input [tns:stopCDRTransmission](#)  
output [tns:stopCDRTransmissionResponse](#)  
fault [tns:negResponse](#)

### CSTA-Stop-CDR-Transmission-as-output

output [tns:stopCDRTransmission](#)  
input [tns:stopCDRTransmissionResponse](#)  
fault [tns:negResponse](#)

source

```
<operation name="CSTA-CDR-Notification">
  <output message="tns:cDRNotification"/>
  <input message="tns:cDRNotificationResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-CDR-Report">
  <output message="tns:cDRReport"/>
  <input message="tns:cDRReportResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Send-Stored-CDR">
  <input message="tns:sendStoredCDR"/>
  <output message="tns:sendStoredCDRResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

```
<operation name="CSTA-Start-CDR-Transmission">
  <input message="tns:startCDRTransmission"/>
  <output message="tns:startCDRTransmissionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>

<operation name="CSTA-Stop-CDR-Transmission-as-input">
  <input message="tns:stopCDRTransmission"/>
  <output message="tns:stopCDRTransmissionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>

<operation name="CSTA-Stop-CDR-Transmission-as-output">
  <output message="tns:stopCDRTransmission"/>
  <input message="tns:stopCDRTransmissionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
```

## 22 Vendor Specific Features - Messages and Operations

This clause specifies the WSDL messages and operations for the Vendor Specific features.

### 22.1 Messages

#### message **escapeRegister**

```
parts    parameter
         element    csta:EscapeRegister
source   <message name="escapeRegister">
         <part name="parameter" element="csta:EscapeRegister"/>
         </message>
```

#### message **escapeRegisterResponse**

```
parts    parameter
         element    csta:EscapeRegisterResponse
source   <message name="escapeRegisterResponse">
         <part name="parameter" element="csta:EscapeRegisterResponse"/>
         </message>
```

#### message **escapeRegisterAbort**

```
parts    parameter
         element    csta:EscapeRegisterAbort
source   <message name="escapeRegisterAbort">
         <part name="parameter" element="csta:EscapeRegisterAbort"/>
         </message>
```

#### message **escapeRegisterCancel**

```
parts    parameter
         element    csta:EscapeRegisterCancel
source   <message name="escapeRegisterCancel">
         <part name="parameter" element="csta:EscapeRegisterCancel"/>
         </message>
```



### message **escapeRegisterCancelResponse**

```
parts    parameter
         element  csta:EscapeRegisterCancelResponse
source  <message name="escapeRegisterCancelResponse">
         <part name="parameter" element="csta:EscapeRegisterCancelResponse"/>
         </message>
```

### message **escape**

```
parts    parameter
         element  csta:Escape
source  <message name="escape">
         <part name="parameter" element="csta:Escape"/>
         </message>
```

### message **escapeResponse**

```
parts    parameter
         element  csta:EscapeResponse
source  <message name="escapeResponse">
         <part name="parameter" element="csta:EscapeResponse"/>
         </message>
```

### message **privateDataVersionSelection**

```
parts    parameter
         element  csta:PrivateDataVersionSelection
source  <message name="privateDataVersionSelection">
         <part name="parameter" element="csta:PrivateDataVersionSelection"/>
         </message>
```

### message **privateDataVersionSelectionResponse**

```
parts    parameter
         element  csta:PrivateDataVersionSelectionResponse
source  <message name="privateDataVersionSelectionResponse">
         <part name="parameter" element="csta:PrivateDataVersionSelectionResponse"/>
         </message>
```

message **privateEvent**

parts	<b>parameter</b>
	element <b>csta:PrivateEvent</b>
source	<code>&lt;message name="privateEvent"&gt;</code> <code>&lt;part name="parameter" element="csta:PrivateEvent"/&gt;</code> <code>&lt;/message&gt;</code>

## 22.2 Operations

operations	<p><b>CSTA-EscapeRegister</b></p> <p>input <a href="#">tns:escapeRegister</a></p> <p>output <a href="#">tns:escapeRegisterResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Escape-Register-Abort</b></p> <p>output <a href="#">tns:escapeRegisterAbort</a></p> <p><b>CSTA-Escape-Register-Cancel</b></p> <p>input <a href="#">tns:escapeRegisterCancel</a></p> <p>output <a href="#">tns:escapeRegisterCancelResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Escape-as-input</b></p> <p>input <a href="#">tns:escape</a></p> <p>output <a href="#">tns:escapeResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Escape-as-output</b></p> <p>output <a href="#">tns:escape</a></p> <p>input <a href="#">tns:escapeResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Private-Data-Version-Selection</b></p> <p>input <a href="#">tns:privateDataVersionSelection</a></p> <p>output <a href="#">tns:privateDataVersionSelectionResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Private-Event</b></p> <p>output <a href="#">tns:privateEvent</a></p>
source	<pre> &lt;operation name="CSTA-EscapeRegister"&gt;   &lt;input message="tns:escapeRegister"/&gt;   &lt;output message="tns:escapeRegisterResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Escape-Register-Abort"&gt;   &lt;output message="tns:escapeRegisterAbort"/&gt; &lt;/operation&gt; &lt;operation name="CSTA-Escape-Register-Cancel"&gt;   &lt;input message="tns:escapeRegisterCancel"/&gt;   &lt;output message="tns:escapeRegisterCancelResponse"/&gt;   &lt;fault name="FaultName" message="tns:negResponse"/&gt; &lt;/operation&gt; </pre>

```
<operation name="CSTA-Escape-as-input">
  <input message="tns:escape"/>
  <output message="tns:escapeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Escape-as-output">
  <output message="tns:escape"/>
  <input message="tns:escapeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Private-Data-Version-Selection">
  <input message="tns:privateDataVersionSelection"/>
  <output message="tns:privateDataVersionSelectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Private-Event">
  <output message="tns:privateEvent"/>
</operation>
```

## 23 Basic Telephony Profile – Port Type

This clause specifies the CSTA-Basic-Telephony-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Basic Telephony profile as specified in ECMA-269 2.1.3.1.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Basic-Telephony-Profile**

operations	<p><b>CSTA-Get-Switching-Function-Capabilities</b></p> <p>input <a href="#"><u>tns:getSwitchingFunctionCaps</u></a></p> <p>output <a href="#"><u>tns:getSwitchingFunctionCapsResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a></p> <p>output <a href="#"><u>tns:answerCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a></p> <p>output <a href="#"><u>tns:clearConnectionResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#"><u>tns:makeCall</u></a></p> <p>output <a href="#"><u>tns:makeCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Start</b></p> <p>input <a href="#"><u>tns:monitorStart</u></a></p> <p>output <a href="#"><u>tns:monitorStartResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Stop-as-input</b></p> <p>input <a href="#"><u>tns:monitorStop</u></a></p> <p>output <a href="#"><u>tns:monitorStopResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Connection-Cleared-event</b></p> <p>output <a href="#"><u>tns:connectionClearedEvent</u></a></p> <p><b>CSTA-Delivered-event</b></p> <p>output <a href="#"><u>tns:deliveredEvent</u></a></p>
------------	---

**CSTA-Established-event**output [tns:establishedEvent](#)**CSTA-Failed-event**output [tns:failedEvent](#)**CSTA-Network-Reached-event**output [tns:networkReachedEvent](#)**CSTA-Originated-event**output [tns:originatedEvent](#)**CSTA-Service-Initiated-event**output [tns:serviceInitiatedEvent](#)

```
source <portType name="CSTA-Basic-Telephony-Profile">
  <operation name="CSTA-Get-Switching-Function-Capabilities">
    <input message="tns:getSwitchingFunctionCaps"/>
    <output message="tns:getSwitchingFunctionCapsResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Make-Call">
    <input message="tns:makeCall"/>
    <output message="tns:makeCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Start">
    <input message="tns:monitorStart"/>
    <output message="tns:monitorStartResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Stop-as-input">
    <input message="tns:monitorStop"/>
    <output message="tns:monitorStopResponse"/>
  </operation>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
</portType>
```

## 24 Routing Profile - Port Type

This clause specifies the CSTA-Routeing-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Routeing profile as specified in ECMA-269 2.1.3.2.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Routeing-Profile**

operations	<p><b>CSTA-Route-Request</b></p> <p>output <a href="#">tns:routeRequest</a></p>
	<p><b>CSTA-Route-Select</b></p> <p>input <a href="#">tns:routeSelect</a></p>
	<p><b>CSTA-Route-End-as-output</b></p> <p>output <a href="#">tns:routeEnd</a></p>
source	<pre> &lt;portType name="CSTA-Routeing-Profile"&gt;   &lt;operation name="CSTA-Route-Request"&gt;     &lt;output message="tns:routeRequest"/&gt;   &lt;/operation&gt;   &lt;operation name="CSTA-Route-Select"&gt;     &lt;input message="tns:routeSelect"/&gt;   &lt;/operation&gt;   &lt;operation name="CSTA-Route-End-as-output"&gt;     &lt;output message="tns:routeEnd"/&gt;   &lt;/operation&gt; &lt;/portType&gt; </pre>



## 25 Level 1a Voice Browser Profile - Port Type

This clause specifies the CSTA-Level-1a-Voice-Browser-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Level 1a Voice Browser profile as specified in ECMA-269 2.1.3.3.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Level-1a-Voice-Browser-Profile**

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a></p> <p>output <a href="#"><u>tns:answerCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a></p> <p>output <a href="#"><u>tns:clearConnectionResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Single-Step-Transfer-Call</b></p> <p>input <a href="#"><u>tns:singleStepTransferCall</u></a></p> <p>output <a href="#"><u>tns:singleStepTransferCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Start</b></p> <p>input <a href="#"><u>tns:monitorStart</u></a></p> <p>output <a href="#"><u>tns:monitorStartResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Stop-as-input</b></p> <p>input <a href="#"><u>tns:monitorStop</u></a></p> <p>output <a href="#"><u>tns:monitorStopResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Connection-Cleared-event</b></p> <p>output <a href="#"><u>tns:connectionClearedEvent</u></a></p> <p><b>CSTA-Failed-event</b></p> <p>output <a href="#"><u>tns:failedEvent</u></a></p> <p><b>CSTA-Delivered-event</b></p> <p>output <a href="#"><u>tns:deliveredEvent</u></a></p> <p><b>CSTA-Transferred-event</b></p> <p>output <a href="#"><u>tns:transferredEvent</u></a></p>
------------	---

### CSTA-Established-event

output [tns:establishedEvent](#)

```
source <portType name="CSTA-Level-1a-Voice-Browser-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Single-Step-Transfer-Call">
    <input message="tns:singleStepTransferCall"/>
    <output message="tns:singleStepTransferCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Start">
    <input message="tns:monitorStart"/>
    <output message="tns:monitorStartResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Stop-as-input">
    <input message="tns:monitorStop"/>
    <output message="tns:monitorStopResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Connection-Cleared-event">
    <output message="tns:connectionClearedEvent"/>
  </operation>
  <operation name="CSTA-Failed-event">
    <output message="tns:failedEvent"/>
  </operation>
  <operation name="CSTA-Delivered-event">
    <output message="tns:deliveredEvent"/>
  </operation>
  <operation name="CSTA-Transferred-event">
    <output message="tns:transferredEvent"/>
  </operation>
  <operation name="CSTA-Established-event">
    <output message="tns:establishedEvent"/>
  </operation>
</portType>
```

## 26 Level 1b Voice Browser Profile - Port Type

This clause specifies the CSTA-Level-1b-Voice-Browser-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Level 1b Voice Browser profile as specified in ECMA-269 2.1.3.4.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Level-1b-Voice-Browser-Profile**

operations	<b>CSTA-Answer-Call</b>
	input <a href="#"><u>tns:answerCall</u></a>
	output <a href="#"><u>tns:answerCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Clear-Connection</b>
	input <a href="#"><u>tns:clearConnection</u></a>
	output <a href="#"><u>tns:clearConnectionResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Deflect-Call</b>
	input <a href="#"><u>tns:deflectCall</u></a>
	output <a href="#"><u>tns:deflectCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Monitor-Start</b>
	input <a href="#"><u>tns:monitorStart</u></a>
	output <a href="#"><u>tns:monitorStartResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Monitor-Stop-as-input</b>
	input <a href="#"><u>tns:monitorStop</u></a>
	output <a href="#"><u>tns:monitorStopResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Connection-Cleared-event</b>
	output <a href="#"><u>tns:connectionClearedEvent</u></a>
	<b>CSTA-Failed-event</b>
	output <a href="#"><u>tns:failedEvent</u></a>
	<b>CSTA-Delivered-event</b>
	output <a href="#"><u>tns:deliveredEvent</u></a>
	<b>CSTA-Diverted-event</b>
	output <a href="#"><u>tns:divertedEvent</u></a>

### CSTA-Established-event

output [tns:establishedEvent](#)

```
source <portType name="CSTA-Level-1b-Voice-Browser-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Deflect-Call">
    <input message="tns:deflectCall"/>
    <output message="tns:deflectCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Start">
    <input message="tns:monitorStart"/>
    <output message="tns:monitorStartResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Stop-as-input">
    <input message="tns:monitorStop"/>
    <output message="tns:monitorStopResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Connection-Cleared-event">
    <output message="tns:connectionClearedEvent"/>
  </operation>
  <operation name="CSTA-Failed-event">
    <output message="tns:failedEvent"/>
  </operation>
  <operation name="CSTA-Delivered-event">
    <output message="tns:deliveredEvent"/>
  </operation>
  <operation name="CSTA-Diverted-event">
    <output message="tns:divertedEvent"/>
  </operation>
  <operation name="CSTA-Established-event">
    <output message="tns:establishedEvent"/>
  </operation>
</portType>
```

## 27 Level 2a Voice Browser Profile - Port Type

This clause specifies the CSTA-Level-2a-Voice-Browser-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Level 2a Voice Browser profile as specified in ECMA-269 2.1.3.5.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype CSTA-Level-2a-Voice-Browser-Profile

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#">tns:answerCall</a></p> <p>output <a href="#">tns:answerCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#">tns:clearConnection</a></p> <p>output <a href="#">tns:clearConnectionResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#">tns:makeCall</a></p> <p>output <a href="#">tns:makeCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Single-Step-Transfer-Call</b></p> <p>input <a href="#">tns:singleStepTransferCall</a></p> <p>output <a href="#">tns:singleStepTransferCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Monitor-Start</b></p> <p>input <a href="#">tns:monitorStart</a></p> <p>output <a href="#">tns:monitorStartResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Monitor-Stop-as-input</b></p> <p>input <a href="#">tns:monitorStop</a></p> <p>output <a href="#">tns:monitorStopResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Connection-Cleared-event</b></p> <p>output <a href="#">tns:connectionClearedEvent</a></p> <p><b>CSTA-Delivered-event</b></p> <p>output <a href="#">tns:deliveredEvent</a></p> <p><b>CSTA-Established-event</b></p> <p>output <a href="#">tns:establishedEvent</a></p>
------------	---

#### CSTA-Failed-event

output [tns:failedEvent](#)

#### CSTA-Network-Reached-event

output [tns:networkReachedEvent](#)

#### CSTA-Originated-event

output [tns:originatedEvent](#)

#### CSTA-Transferred-event

output [tns:transferredEvent](#)

```
source <portType name="CSTA-Level-2a-Voice-Browser-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Make-Call">
    <input message="tns:makeCall"/>
    <output message="tns:makeCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Single-Step-Transfer-Call">
    <input message="tns:singleStepTransferCall"/>
    <output message="tns:singleStepTransferCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Start">
    <input message="tns:monitorStart"/>
    <output message="tns:monitorStartResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Stop-as-input">
    <input message="tns:monitorStop"/>
    <output message="tns:monitorStopResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Connection-Cleared-event">
```

```
<output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
</portType>
```

## 28 Level 2b Voice Browser Profile - Port Type

This clause specifies the CSTA-Level-2b-Voice-Browser-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Level 2b Voice Browser profile as specified in ECMA-269 2.1.3.5.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype CSTA-Level-2b-Voice-Browser-Profile

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a></p> <p>output <a href="#"><u>tns:answerCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a></p> <p>output <a href="#"><u>tns:clearConnectionResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#"><u>tns:makeCall</u></a></p> <p>output <a href="#"><u>tns:makeCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#"><u>tns:deflectCall</u></a></p> <p>output <a href="#"><u>tns:deflectCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Start</b></p> <p>input <a href="#"><u>tns:monitorStart</u></a></p> <p>output <a href="#"><u>tns:monitorStartResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Monitor-Stop-as-input</b></p> <p>input <a href="#"><u>tns:monitorStop</u></a></p> <p>output <a href="#"><u>tns:monitorStopResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Connection-Cleared-event</b></p> <p>output <a href="#"><u>tns:connectionClearedEvent</u></a></p> <p><b>CSTA-Delivered-event</b></p> <p>output <a href="#"><u>tns:deliveredEvent</u></a></p>
------------	--



**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Originated-event**

output [tns:originatedEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

```
source <portType name="CSTA-Level-2b-Voice-Browser-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Make-Call">
    <input message="tns:makeCall"/>
    <output message="tns:makeCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Deflect-Call">
    <input message="tns:deflectCall"/>
    <output message="tns:deflectCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Start">
    <input message="tns:monitorStart"/>
    <output message="tns:monitorStartResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Monitor-Stop-as-input">
    <input message="tns:monitorStop"/>
    <output message="tns:monitorStopResponse"/>
  </operation>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
</portType>
```

## 29 Minimal uaCSTA Call Control Profile – Port Type

This clause specifies the CSTA-Minimal-uaCSTA-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Minimal uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.6.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype CSTA-Minimal-uaCSTA-Call-Control-Profile

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#">tns:answerCall</a></p> <p>output <a href="#">tns:answerCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#">tns:clearConnection</a></p> <p>output <a href="#">tns:clearConnectionResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#">tns:deflectCall</a></p> <p>output <a href="#">tns:deflectCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Hold-Call</b></p> <p>input <a href="#">tns:holdCall</a></p> <p>output <a href="#">tns:holdCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#">tns:makeCall</a></p> <p>output <a href="#">tns:makeCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Retrieve-Call</b></p> <p>input <a href="#">tns:retrieveCall</a></p> <p>output <a href="#">tns:retrieveCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Single-Step-Transfer-Call</b></p> <p>input <a href="#">tns:singleStepTransferCall</a></p> <p>output <a href="#">tns:singleStepTransferCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p>
source	<pre>&lt;portType name="CSTA-Minimal-uaCSTA-Call-Control-Profile"&gt;   &lt;operation name="CSTA-Answer-Call"&gt;     &lt;input message="tns:answerCall"/&gt; </pre>

```
<output message="tns:answerCallResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 30 Basic uaCSTA Call Control Profile – Port Type

This clause specifies the CSTA-Basic-uaCSTA-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Basic uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.7.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Basic-uaCSTA-Call-Control-Profile**

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a>  output <a href="#"><u>tns:answerCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a>  output <a href="#"><u>tns:clearConnectionResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#"><u>tns:deflectCall</u></a>  output <a href="#"><u>tns:deflectCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Hold-Call</b></p> <p>input <a href="#"><u>tns:holdCall</u></a>  output <a href="#"><u>tns:holdCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#"><u>tns:makeCall</u></a>  output <a href="#"><u>tns:makeCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Retrieve-Call</b></p> <p>input <a href="#"><u>tns:retrieveCall</u></a>  output <a href="#"><u>tns:retrieveCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Single-Step-Transfer-Call</b></p> <p>input <a href="#"><u>tns:singleStepTransferCall</u></a>  output <a href="#"><u>tns:singleStepTransferCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Connection-Cleared-event</b></p> <p>output <a href="#"><u>tns:connectionClearedEvent</u></a></p>
------------	---

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

**CSTA-Monitor-Start**

input [tns:monitorStart](#)

output [tns:monitorStartResponse](#)

fault [tns:negResponse](#)

**CSTA-Monitor-Stop-as-input**

input [tns:monitorStop](#)

output [tns:monitorStopResponse](#)

fault [tns:negResponse](#)

```
source <portType name="CSTA-Basic-uaCSTA-Call-Control-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
```

```
<output message="tns:clearConnectionResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
```

```
<output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```



## 31 Advanced uaCSTA Call Control Profile – Port Type

This clause specifies the CSTA-Advanced-uaCSTA-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Advanced uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.8.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype CSTA-Advanced-uaCSTA-Call-Control-Profile

operations	<p><b>CSTA-Alternate-Call</b></p> <p>input <a href="#">tns:alternateCall</a></p> <p>output <a href="#">tns:alternateCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Answer-Call</b></p> <p>input <a href="#">tns:answerCall</a></p> <p>output <a href="#">tns:answerCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#">tns:clearConnection</a></p> <p>output <a href="#">tns:clearConnectionResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Consultation-Call</b></p> <p>input <a href="#">tns:consultationCall</a></p> <p>output <a href="#">tns:consultationCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#">tns:deflectCall</a></p> <p>output <a href="#">tns:deflectCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Hold-Call</b></p> <p>input <a href="#">tns:holdCall</a></p> <p>output <a href="#">tns:holdCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#">tns:makeCall</a></p> <p>output <a href="#">tns:makeCallResponse</a></p> <p>fault <a href="#">tns:negResponse</a></p>
------------	---

#### CSTA-Reconnect-Call

input [tns:reconnectCall](#)  
output [tns:reconnectCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Retrieve-Call

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Single-Step-Transfer-Call

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Transfer-Call

input [tns:transferCall](#)  
output [tns:transferCallResponse](#)  
fault [tns:negResponse](#)

#### CSTA-Connection-Cleared-event

output [tns:connectionClearedEvent](#)

#### CSTA-Delivered-event

output [tns:deliveredEvent](#)

#### CSTA-Diverted-event

output [tns:divertedEvent](#)

#### CSTA-Established-event

output [tns:establishedEvent](#)

#### CSTA-Failed-event

output [tns:failedEvent](#)

#### CSTA-Held-event

output [tns:heldEvent](#)

#### CSTA-Network-Reached-event

output [tns:networkReachedEvent](#)

#### CSTA-Originated-event

output [tns:originatedEvent](#)

#### CSTA-Retrieved-event

output [tns:retrievedEvent](#)

#### CSTA-Service-Initiated-event

output [tns:serviceInitiatedEvent](#)

#### CSTA-Transferred-event

output [tns:transferredEvent](#)

#### CSTA-Monitor-Start

input [tns:monitorStart](#)

output [tns:monitorStartResponse](#)

fault [tns:negResponse](#)

#### CSTA-Monitor-Stop-as-input

input [tns:monitorStop](#)

output [tns:monitorStopResponse](#)

fault [tns:negResponse](#)

#### CSTA-Get-Switching-Function-Capabilities

input [tns:getSwitchingFunctionCaps](#)

output [tns:getSwitchingFunctionCapsResponse](#)

fault [tns:negResponse](#)

```
source <portType name="CSTA-Advanced-uaCSTA-Call-Control-Profile">
  <operation name="CSTA-Alternate-Call">
    <input message="tns:alternateCall"/>
    <output message="tns:alternateCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Consultation-Call">
    <input message="tns:consultationCall"/>
    <output message="tns:consultationCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Deflect-Call">
```

```
<input message="tns:deflectCall"/>
<output message="tns:deflectCallResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reconnect-Call">
  <input message="tns:reconnectCall"/>
  <output message="tns:reconnectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Transfer-Call">
  <input message="tns:transferCall"/>
  <output message="tns:transferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
```

```
<output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Switching-Function-Capabilities">
  <input message="tns:getSwitchingFunctionCaps"/>
  <output message="tns:getSwitchingFunctionCapsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 32 Conferencing uaCSTA with Basic Call Control Profile – Port Type

This clause specifies the CSTA-Conferencing-uaCSTA-Basic-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Conferencing uaCSTA Profile with the Basic uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.9.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Conferencing-uaCSTA-Basic-Call-Control-Profile**

operations	<p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a></p> <p>output <a href="#"><u>tns:answerCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a></p> <p>output <a href="#"><u>tns:clearConnectionResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#"><u>tns:deflectCall</u></a></p> <p>output <a href="#"><u>tns:deflectCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Hold-Call</b></p> <p>input <a href="#"><u>tns:holdCall</u></a></p> <p>output <a href="#"><u>tns:holdCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Make-Call</b></p> <p>input <a href="#"><u>tns:makeCall</u></a></p> <p>output <a href="#"><u>tns:makeCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Retrieve-Call</b></p> <p>input <a href="#"><u>tns:retrieveCall</u></a></p> <p>output <a href="#"><u>tns:retrieveCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Single-Step-Transfer-Call</b></p> <p>input <a href="#"><u>tns:singleStepTransferCall</u></a></p> <p>output <a href="#"><u>tns:singleStepTransferCallResponse</u></a></p> <p>fault <a href="#"><u>tns:negResponse</u></a></p>
------------	---

**CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

**CSTA-Monitor-Start**

input [tns:monitorStart](#)

output [tns:monitorStartResponse](#)

fault [tns:negResponse](#)

**CSTA-Monitor-Stop-as-input**

input [tns:monitorStop](#)

output [tns:monitorStopResponse](#)

fault [tns:negResponse](#)

```
source <portType name="CSTA-Conferencing-uaCSTA-Basic-Call-Control-Profile">
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
</portType>
```

```
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
```



```
<output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

### 33 Conferencing uaCSTA with Advanced Call Control Profile – Port Type

This clause specifies the CSTA-Conferencing-uaCSTA-Advanced-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Conferencing uaCSTA Call Control Profile with the Advanced uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.9.

Implementations may support other operations in addition to the operations specified in this port type.

#### porttype **CSTA-Conferencing-uaCSTA-Advanced-Call-Control-Profile**

operations

input [tns:alternateCall](#)  
 output [tns:alternateCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Answer-Call**

input [tns:answerCall](#)  
 output [tns:answerCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Clear-Connection**

input [tns:clearConnection](#)  
 output [tns:clearConnectionResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Conference-Call**

input [tns:conferenceCall](#)  
 output [tns:conferenceCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Consultation-Call**

input [tns:consultationCall](#)  
 output [tns:consultationCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Deflect-Call**

input [tns:deflectCall](#)  
 output [tns:deflectCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Hold-Call**

input [tns:holdCall](#)  
 output [tns:holdCallResponse](#)  
 fault [tns:negResponse](#)

#### **CSTA-Make-Call**

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Reconnect-Call**

input [tns:reconnectCall](#)  
output [tns:reconnectCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Single-Step-Conference-Call**

input [tns:singleStepConferenceCall](#)  
output [tns:singleStepConferenceCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Transfer-Call**

input [tns:transferCall](#)  
output [tns:transferCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Conferenced-event**

output [tns:conferencedEvent](#)

#### **CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

#### **CSTA-Delivered-event**

output [tns:deliveredEvent](#)

#### **CSTA-Diverted-event**

output [tns:divertedEvent](#)

#### **CSTA-Established-event**

output [tns:establishedEvent](#)

#### CSTA-Failed-event

output [tns:failedEvent](#)

#### CSTA-Held-event

output [tns:heldEvent](#)

#### CSTA-Network-Reached-event

output [tns:networkReachedEvent](#)

#### CSTA-Originated-event

output [tns:originatedEvent](#)

#### CSTA-Retrieved-event

output [tns:retrievedEvent](#)

#### CSTA-Service-Initiated-event

output [tns:serviceInitiatedEvent](#)

#### CSTA-Transferred-event

output [tns:transferredEvent](#)

#### CSTA-Monitor-Start

input [tns:monitorStart](#)  
 output [tns:monitorStartResponse](#)  
 fault [tns:negResponse](#)

#### CSTA-Monitor-Stop-as-input

input [tns:monitorStop](#)  
 output [tns:monitorStopResponse](#)  
 fault [tns:negResponse](#)

#### CSTA-Get-Switching-Function-Capabilities

input [tns:getSwitchingFunctionCaps](#)  
 output [tns:getSwitchingFunctionCapsResponse](#)  
 fault [tns:negResponse](#)

```
source <portType name="CSTA-Conferencing-uaCSTA-Advanced-Call-Control-Profile">
  <operation name="CSTA-Conferencing-Advanced-Call-Control-Profile">
    <input message="tns:alternateCall"/>
    <output message="tns:alternateCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
  </operation>
</portType>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Conference-Call">
  <input message="tns:conferenceCall"/>
  <output message="tns:conferenceCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Consultation-Call">
  <input message="tns:consultationCall"/>
  <output message="tns:consultationCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reconnect-Call">
  <input message="tns:reconnectCall"/>
  <output message="tns:reconnectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Conference-Call">
  <input message="tns:singleStepConferenceCall"/>
```

```
<output message="tns:singleStepConferenceCallResponse"/>
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Transfer-Call">
  <input message="tns:transferCall"/>
  <output message="tns:transferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Conferenced-event">
  <output message="tns:conferencedEvent"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
  <output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
```

```
<output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Switching-Function-Capabilities">
  <input message="tns:getSwitchingFunctionCaps"/>
  <output message="tns:getSwitchingFunctionCapsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 34 Basic uaCSTA Device Feature with Basic Call Control Profile – Port Type

This clause specifies the CSTA-Basic-uaCSTA-Device-Feature-Basic-Call-Control Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Basic uaCSTA Device Feature Profile with the Basic uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.10.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Basic-uaCSTA-Device-Feature-Basic-Call-Control-Profile**

operations	<b>CSTA-Set-Do-Not-Disturb</b>
	input <a href="#"><u>tns:setDoNotDisturb</u></a>
	output <a href="#"><u>tns:setDoNotDisturbResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Set-Forwarding</b>
	input <a href="#"><u>tns:setForwarding</u></a>
	output <a href="#"><u>tns:setForwardingResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Do-Not-Disturb-Event</b>
	output <a href="#"><u>tns:doNotDisturbEvent</u></a>
	<b>CSTA-Forwarding-Event</b>
	output <a href="#"><u>tns:forwardingEvent</u></a>
	<b>CSTA-Answer-Call</b>
	input <a href="#"><u>tns:answerCall</u></a>
	output <a href="#"><u>tns:answerCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Clear-Connection</b>
	input <a href="#"><u>tns:clearConnection</u></a>
	output <a href="#"><u>tns:clearConnectionResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Deflect-Call</b>
	input <a href="#"><u>tns:deflectCall</u></a>
	output <a href="#"><u>tns:deflectCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Hold-Call</b>
	input <a href="#"><u>tns:holdCall</u></a>
	output <a href="#"><u>tns:holdCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>



**CSTA-Make-Call**

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

### CSTA-Monitor-Start

input [tns:monitorStart](#)  
output [tns:monitorStartResponse](#)  
fault [tns:negResponse](#)

### CSTA-Monitor-Stop-as-input

input [tns:monitorStop](#)  
output [tns:monitorStopResponse](#)  
fault [tns:negResponse](#)

```
source <portType name="CSTA-Basic-uaCSTA-Device-Feature-Basic-Call-Control-Profile">
  <operation name="CSTA-Set-Do-Not-Disturb">
    <input message="tns:setDoNotDisturb"/>
    <output message="tns:setDoNotDisturbResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Set-Forwarding">
    <input message="tns:setForwarding"/>
    <output message="tns:setForwardingResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Do-Not-Disturb-Event">
    <output message="tns:doNotDisturbEvent"/>
  </operation>
  <operation name="CSTA-Forwarding-Event">
    <output message="tns:forwardingEvent"/>
  </operation>
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Deflect-Call">
    <input message="tns:deflectCall"/>
    <output message="tns:deflectCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Hold-Call">
    <input message="tns:holdCall"/>
    <output message="tns:holdCallResponse"/>
  </operation>
</portType>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
```

```
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 35 Basic uaCSTA Device Feature with Advanced Call Control Profile – Port Type

This clause specifies the CSTA-Basic-uaCSTA-Device-Feature-Advanced-Call-Control Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Basic uaCSTA Device Feature Profile with the Advanced uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.10.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Basic-uaCSTA-Device-Feature-Advanced-Call-Control-Profile**

operations	<b>CSTA-Set-Do-Not-Disturb</b>
	input <a href="#"><u>tns:setDoNotDisturb</u></a>
	output <a href="#"><u>tns:setDoNotDisturbResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Set-Forwarding</b>
	input <a href="#"><u>tns:setForwarding</u></a>
	output <a href="#"><u>tns:setForwardingResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Do-Not-Disturb-Event</b>
	output <a href="#"><u>tns:doNotDisturbEvent</u></a>
	<b>CSTA-Forwarding-Event</b>
	output <a href="#"><u>tns:forwardingEvent</u></a>
	<b>CSTA-Alternate-Call</b>
	input <a href="#"><u>tns:alternateCall</u></a>
	output <a href="#"><u>tns:alternateCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Answer-Call</b>
	input <a href="#"><u>tns:answerCall</u></a>
	output <a href="#"><u>tns:answerCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Clear-Connection</b>
	input <a href="#"><u>tns:clearConnection</u></a>
	output <a href="#"><u>tns:clearConnectionResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Consultation-Call</b>
	input <a href="#"><u>tns:consultationCall</u></a>
	output <a href="#"><u>tns:consultationCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>

**CSTA-Deflect-Call**

input [tns:deflectCall](#)  
output [tns:deflectCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Hold-Call**

input [tns:holdCall](#)  
output [tns:holdCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Make-Call**

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Reconnect-Call**

input [tns:reconnectCall](#)  
output [tns:reconnectCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Transfer-Call**

input [tns:transferCall](#)  
output [tns:transferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Originated-event**

output [tns:originatedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

**CSTA-Monitor-Start**

input [tns:monitorStart](#)  
 output [tns:monitorStartResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Monitor-Stop-as-input**

input [tns:monitorStop](#)  
 output [tns:monitorStopResponse](#)  
 fault [tns:negResponse](#)

**CSTA-Get-Switching-Function-Capabilities**

input [tns:getSwitchingFunctionCaps](#)  
 output [tns:getSwitchingFunctionCapsResponse](#)  
 fault [tns:negResponse](#)

```
source <portType name="CSTA-Basic-uaCSTA-Device-Feature-Advanced-Call-Control-Profile">
  <operation name="CSTA-Set-Do-Not-Disturb">
    <input message="tns:setDoNotDisturb"/>
    <output message="tns:setDoNotDisturbResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
```

```
<operation name="CSTA-Set-Forwarding">
  <input message="tns:setForwarding"/>
  <output message="tns:setForwardingResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Do-Not-Disturb-Event">
  <output message="tns:doNotDisturbEvent"/>
</operation>
<operation name="CSTA-Forwarding-Event">
  <output message="tns:forwardingEvent"/>
</operation>
<operation name="CSTA-Alternate-Call">
  <input message="tns:alternateCall"/>
  <output message="tns:alternateCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Answer-Call">
  <input message="tns:answerCall"/>
  <output message="tns:answerCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Consultation-Call">
  <input message="tns:consultationCall"/>
  <output message="tns:consultationCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
```



```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reconnect-Call">
  <input message="tns:reconnectCall"/>
  <output message="tns:reconnectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Transfer-Call">
  <input message="tns:transferCall"/>
  <output message="tns:transferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
```

```
<output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Switching-Function-Capabilities">
  <input message="tns:getSwitchingFunctionCaps"/>
  <output message="tns:getSwitchingFunctionCapsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 36 Speaker uaCSTA Device Feature with Basic Call Control Profile – Port Type

This clause specifies the CSTA-Speaker-uaCSTA-Device-Feature-Basic-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Speaker uaCSTA Device Feature Profile with the Basic uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.11.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Speaker-uaCSTA-Device-Feature-Basic-Call-Control-Profile**

operations	<p><b>CSTA-Set-Speaker-Mute</b></p> <p>input <a href="#"><u>tns:setSpeakerMute</u></a>  output <a href="#"><u>tns:setSpeakerMuteResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Set-Speaker-Volume</b></p> <p>input <a href="#"><u>tns:setSpeakerVolume</u></a>  output <a href="#"><u>tns:setSpeakerVolumeResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Speaker-Mute-Event</b></p> <p>output <a href="#"><u>tns:speakerMuteEvent</u></a></p> <p><b>CSTA-Speaker-Volume-Event</b></p> <p>output <a href="#"><u>tns:speakerVolumeEvent</u></a></p> <p><b>CSTA-Answer-Call</b></p> <p>input <a href="#"><u>tns:answerCall</u></a>  output <a href="#"><u>tns:answerCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Clear-Connection</b></p> <p>input <a href="#"><u>tns:clearConnection</u></a>  output <a href="#"><u>tns:clearConnectionResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Deflect-Call</b></p> <p>input <a href="#"><u>tns:deflectCall</u></a>  output <a href="#"><u>tns:deflectCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p> <p><b>CSTA-Hold-Call</b></p> <p>input <a href="#"><u>tns:holdCall</u></a>  output <a href="#"><u>tns:holdCallResponse</u></a>  fault <a href="#"><u>tns:negResponse</u></a></p>
------------	---

**CSTA-Make-Call**

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

**CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

**CSTA-Delivered-event**

output [tns:deliveredEvent](#)

**CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

### CSTA-Monitor-Start

input [tns:monitorStart](#)  
 output [tns:monitorStartResponse](#)  
 fault [tns:negResponse](#)

### CSTA-Monitor-Stop-as-input

input [tns:monitorStop](#)  
 output [tns:monitorStopResponse](#)  
 fault [tns:negResponse](#)

```
source <portType name="CSTA-Speaker-uaCSTA-Device-Feature-Basic-Call-Control-Profile">
  <operation name="CSTA-Set-Speaker-Mute">
    <input message="tns:setSpeakerMute"/>
    <output message="tns:setSpeakerMuteResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Set-Speaker-Volume">
    <input message="tns:setSpeakerVolume"/>
    <output message="tns:setSpeakerVolumeResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Speaker-Mute-Event">
    <output message="tns:speakerMuteEvent"/>
  </operation>
  <operation name="CSTA-Speaker-Volume-Event">
    <output message="tns:speakerVolumeEvent"/>
  </operation>
  <operation name="CSTA-Answer-Call">
    <input message="tns:answerCall"/>
    <output message="tns:answerCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Clear-Connection">
    <input message="tns:clearConnection"/>
    <output message="tns:clearConnectionResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Deflect-Call">
    <input message="tns:deflectCall"/>
    <output message="tns:deflectCallResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
  <operation name="CSTA-Hold-Call">
    <input message="tns:holdCall"/>
    <output message="tns:holdCallResponse"/>
  </operation>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
```

```
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```

## 37 Speaker uaCSTA Device Feature with Advanced Call Control Profile – Port Type

This clause specifies the CSTA-Speaker-uaCSTA-Device-Feature-Advanced-Call-Control-Profile portType. This portType specifies the required CSTA WSDL operations to support the CSTA Speaker uaCSTA Device Feature Profile with the Advanced uaCSTA Call Control Profile as specified in ECMA-269 2.1.3.11.

Implementations may support other operations in addition to the operations specified in this port type.

### porttype **CSTA-Speaker-uaCSTA-Device-Feature-Advanced-Call-Control-Profile**

operations	<b>CSTA-Set-Speaker-Mute</b>
	input <a href="#"><u>tns:setSpeakerMute</u></a>
	output <a href="#"><u>tns:setSpeakerMuteResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Set-Speaker-Volume</b>
	input <a href="#"><u>tns:setSpeakerVolume</u></a>
	output <a href="#"><u>tns:setSpeakerVolumeResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Speaker-Mute-Event</b>
	output <a href="#"><u>tns:speakerMuteEvent</u></a>
	<b>CSTA-Speaker-Volume-Event</b>
	output <a href="#"><u>tns:speakerVolumeEvent</u></a>
	<b>CSTA-Alternate-Call</b>
	input <a href="#"><u>tns:alternateCall</u></a>
	output <a href="#"><u>tns:alternateCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Answer-Call</b>
	input <a href="#"><u>tns:answerCall</u></a>
	output <a href="#"><u>tns:answerCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Clear-Connection</b>
	input <a href="#"><u>tns:clearConnection</u></a>
	output <a href="#"><u>tns:clearConnectionResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>
	<b>CSTA-Consultation-Call</b>
	input <a href="#"><u>tns:consultationCall</u></a>
	output <a href="#"><u>tns:consultationCallResponse</u></a>
	fault <a href="#"><u>tns:negResponse</u></a>



#### **CSTA-Deflect-Call**

input [tns:deflectCall](#)  
output [tns:deflectCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Hold-Call**

input [tns:holdCall](#)  
output [tns:holdCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Make-Call**

input [tns:makeCall](#)  
output [tns:makeCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Reconnect-Call**

input [tns:reconnectCall](#)  
output [tns:reconnectCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Retrieve-Call**

input [tns:retrieveCall](#)  
output [tns:retrieveCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Single-Step-Transfer-Call**

input [tns:singleStepTransferCall](#)  
output [tns:singleStepTransferCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Transfer-Call**

input [tns:transferCall](#)  
output [tns:transferCallResponse](#)  
fault [tns:negResponse](#)

#### **CSTA-Connection-Cleared-event**

output [tns:connectionClearedEvent](#)

#### **CSTA-Delivered-event**

output [tns:deliveredEvent](#)

#### **CSTA-Diverted-event**

output [tns:divertedEvent](#)

**CSTA-Established-event**

output [tns:establishedEvent](#)

**CSTA-Failed-event**

output [tns:failedEvent](#)

**CSTA-Held-event**

output [tns:heldEvent](#)

**CSTA-Network-Reached-event**

output [tns:networkReachedEvent](#)

**CSTA-Originated-event**

output [tns:originatedEvent](#)

**CSTA-Retrieved-event**

output [tns:retrievedEvent](#)

**CSTA-Service-Initiated-event**

output [tns:serviceInitiatedEvent](#)

**CSTA-Transferred-event**

output [tns:transferredEvent](#)

**CSTA-Monitor-Start**

input [tns:monitorStart](#)  
output [tns:monitorStartResponse](#)  
fault [tns:negResponse](#)

**CSTA-Monitor-Stop-as-input**

input [tns:monitorStop](#)  
output [tns:monitorStopResponse](#)  
fault [tns:negResponse](#)

**CSTA-Get-Switching-Function-Capabilities**

input [tns:getSwitchingFunctionCaps](#)  
output [tns:getSwitchingFunctionCapsResponse](#)  
fault [tns:negResponse](#)

```
source <portType name="CSTA-Speaker-uaCSTA-Device-Feature-Advanced-Call-Control-Profile">
  <operation name="CSTA-Set-Speaker-Mute">
    <input message="tns:setSpeakerMute"/>
    <output message="tns:setSpeakerMuteResponse"/>
    <fault name="FaultName" message="tns:negResponse"/>
  </operation>
```

```
<operation name="CSTA-Set-Speaker-Volume">
  <input message="tns:setSpeakerVolume"/>
  <output message="tns:setSpeakerVolumeResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Speaker-Mute-Event">
  <output message="tns:speakerMuteEvent"/>
</operation>
<operation name="CSTA-Speaker-Volume-Event">
  <output message="tns:speakerVolumeEvent"/>
</operation>
<operation name="CSTA-Alternate-Call">
  <input message="tns:alternateCall"/>
  <output message="tns:alternateCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Answer-Call">
  <input message="tns:answerCall"/>
  <output message="tns:answerCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Clear-Connection">
  <input message="tns:clearConnection"/>
  <output message="tns:clearConnectionResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Consultation-Call">
  <input message="tns:consultationCall"/>
  <output message="tns:consultationCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Deflect-Call">
  <input message="tns:deflectCall"/>
  <output message="tns:deflectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Hold-Call">
  <input message="tns:holdCall"/>
  <output message="tns:holdCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Make-Call">
  <input message="tns:makeCall"/>
  <output message="tns:makeCallResponse"/>
```

```
<fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Reconnect-Call">
  <input message="tns:reconnectCall"/>
  <output message="tns:reconnectCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Retrieve-Call">
  <input message="tns:retrieveCall"/>
  <output message="tns:retrieveCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Single-Step-Transfer-Call">
  <input message="tns:singleStepTransferCall"/>
  <output message="tns:singleStepTransferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Transfer-Call">
  <input message="tns:transferCall"/>
  <output message="tns:transferCallResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Connection-Cleared-event">
  <output message="tns:connectionClearedEvent"/>
</operation>
<operation name="CSTA-Delivered-event">
  <output message="tns:deliveredEvent"/>
</operation>
<operation name="CSTA-Diverted-event">
  <output message="tns:divertedEvent"/>
</operation>
<operation name="CSTA-Established-event">
  <output message="tns:establishedEvent"/>
</operation>
<operation name="CSTA-Failed-event">
  <output message="tns:failedEvent"/>
</operation>
<operation name="CSTA-Held-event">
  <output message="tns:heldEvent"/>
</operation>
<operation name="CSTA-Network-Reached-event">
  <output message="tns:networkReachedEvent"/>
</operation>
<operation name="CSTA-Originated-event">
```

```
<output message="tns:originatedEvent"/>
</operation>
<operation name="CSTA-Retrieved-event">
  <output message="tns:retrievedEvent"/>
</operation>
<operation name="CSTA-Service-Initiated-event">
  <output message="tns:serviceInitiatedEvent"/>
</operation>
<operation name="CSTA-Transferred-event">
  <output message="tns:transferredEvent"/>
</operation>
<operation name="CSTA-Monitor-Start">
  <input message="tns:monitorStart"/>
  <output message="tns:monitorStartResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Monitor-Stop-as-input">
  <input message="tns:monitorStop"/>
  <output message="tns:monitorStopResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
<operation name="CSTA-Get-Switching-Function-Capabilities">
  <input message="tns:getSwitchingFunctionCaps"/>
  <output message="tns:getSwitchingFunctionCapsResponse"/>
  <fault name="FaultName" message="tns:negResponse"/>
</operation>
</portType>
```



## Annex A (informative)

### Example SOAP 1.1/HTTP Bindings

The following is an example of a WSDL description with SOAP/HTTP bindings. In order to keep the example simple, only one CSTA operation (Answer Call) is included.

```
<definitions
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:tns="http://www.ecma-international.org/standards/ecma-348/csta-wsdl/ed2"
targetNamespace="http://www.ecma-international.org/standards/ecma-348/csta-wsdl/ed2">

  <!--WSDL Type definitions see 6.2-->

  <types>
    <xs:schema targetNamespace="http://www.ecma-international.org/standards/ecma-348/csta-wsdl/ed2">
      <xs:import namespace="http://www.ecma-international.org/standards/ecma-323/csta/ed3" schemaLocation="csta.xsd"/>
    </xs:schema>
  </types>

  <!--WSDL Message definitions see 6.3-->

  <message name="answerCall">
    <part name="parameter" element="csta:AnswerCall"/>
  </message>
  <message name="answerCallResponse">
    <part name="parameter" element="csta:AnswerCallResponse"/>
  </message>
  <message name="negResponse">
    <part name="parameter" type="csta:UniversalFailure"/>
  </message>

  <!--WSDL Operation and PortType definitions see 6.4 and 6.5-->

  <portType name="samplePortType">
    <operation name="CSTA-Answer-Call">
      <input message="tns:answerCall"/>
      <output message="tns:answerCallResponse"/>
      <fault name="FaultName" message="tns:negResponse"/>
    </operation>
  </portType>

  <!--WSDL Binding definitions see 6.6-->

  <binding name="SOAP_HTTP" type="tns:samplePortType">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="CSTA-Answer-Call">
      <soap:operation soapAction="http://CSTA_web_service.com/answerCall" style="document"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
      <fault name="FaultName"/>
    </operation>
  </binding>
</definitions>
```

```
</binding>
```

```
<!--WSDL Port and Service definitions see 6.7 and 6.8-->
```

```
<service name="CSTA_SOAP_Service">
```

```
<port name="CSTA_Sample-Port" binding="tns:SOAP_HTTP">
```

```
<soap:address location="http://CSTA_web_service.com"/>
```

```
</port>
```

```
</service>
```

```
</definitions>
```



## **Annex B** **(informative)**

### **WSDL Considerations**

WSDL version 1.1 has several limitations:

- Does not define fault messages with one-way and Notification operation types
- Does not define concrete SOAP/HTTP bindings for Notification and solicit-response operation types
- A specific operation can only exist in one port type.



## Annex C (informative)

### Summary of Changes in this Edition

The changes included in the 2<sup>nd</sup> edition of ECMA-348 to be compatible with the 3<sup>rd</sup> edition of ECMA-323 and the 6<sup>th</sup> edition of ECMA-269 are:

1. Namespace changes:
  - a. The reference to the ECMA-323 namespace is changed from "<http://www.ecma.ch/standards/ecma-323/csta/ed2>" to "<http://www.ecma-international.org/standards/ecma-323/csta/ed3>"
  - b. The reference to the ECMA-348 namespace is changed from "<http://www.ecma-international.org/standards/ecma-348/csta-wsd/>" to "<http://www.ecma-international.org/standards/ecma-348/csta-wsd/ed2>"
2. The Get CSTA Features service is added to Clause 7.
3. The Partially In Service event is added to Clause 17.
4. Clause 20 – Voice Features
  - a) Clause 20 renamed to Voice Features – Messages and Operations.
  - b) Activate, Clear, Deactivate, Prompt Queue and Start services added to Clause 20.
  - c) Bookmark Reached, Completed, DTMF Detected, Emptied, Interruption Detected, Not Recognized, Recognized, Silence Timeout Expired, Speech Detected, Started, and Voice Error Occurred events added to Clause 20.
5. Port Types added to Clauses 29-37 to reflect the new CSTA Profiles added to ECMA-269.



